DESIGNING A FOUR-TIER ARCHITECTURE FOR A WEB-BASED SIMULATON ENVIRONMENT

Erne Suzila Kassim

UNIVERSITI UTARA MALAYSIA

DESIGNING A FOUR-TIER ARCHITECTURE FOR A WEB-BASED SIMULATON ENVIRONMENT

A thesis submitted to the Graduate School in partial fulfillment of the requirements for the degree Master of Science (Information Technology), Universiti Utara Malaysia

by

Erne Suzila Kassim

Sekolah Siswazah
(Graduate School)
Universiti Utara Malaysia

PERAKUAN KERJA KERTAS PROJEK
(Certification of Project Paper)

Saya, yang bertandatangan, memperakukan bahawa
(I, the undersigned, certify that)

ERNE SUZILA BT. KASSIM

calon untuk Ijazah
(candidate for the degree of) Sarjana Sains (Teknologi Maklumat)

telah mengemukakan kertas projek yang bertajuk
(has presented his/her project paper of the following title)

DESIGNING A FOUR TIER ARCHITECTURE FOR A WEB-BASED SIMULATION ENVIRONMENT

Seperti yang tercatat di muka surat tajuk dan kulit kertas projek
(as it appears on the title page and front cover of project paper)

bahawa kertas projek tersebut boleh diterima dari segi bentuk serta kandungan, dan meliputi bidang ilmu dengan memuaskan.
(that the project paper acceptable in form and content, and that a satisfactory knowledge of the field is covered by the project paper)

Nama Penyelia
(Name of Supervisor) : Prof. Madya Dr. Razman Mat Tahar

Tandatangan
(Signature) :

Tarikh
(Date) : 31 Januari 2001

PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a post-graduate degree from Universiti Utara Malaysia, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by my supervisor or, in his absence, by the Dean of the Graduate School. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Request for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to :

Dean of Graduate School
Universiti Utara Malaysia
06010 Sintok
Kedah Darul Aman

ABSTRAK

Perkembangan teknologi masa kini yang berlaku dengan amat pantas telah mengakibatkan satu evolusi dalam teknologi simulasi. Kewujudan *Internet* dan *World Wide Web* memberi banyak kesan kepada aspek-aspek teknik simulasi yang dilakukan secara konvensional. Ini kerana, teknologi *Web* mempunyai potensi yang besar untuk mengubah cara bagaimana sesuatu model simulasi itu dibangunkan, didokumentasikan, dianalisa dan juga dilaksanakan. Pengaplikasian teknologi *Web* di dalam teknologi simulasi konvensional telah melahirkan satu konsep aplikasi baru, yang dikenali sebagai *Web-based simulation.* Salah satu manfaat yang boleh diambil dari kecanggihan teknologi *Web* adalah melakukan integrasi di antara aplikasi *Web-based simulation* dengan storan data. Antara kelebihan yang nyata apabila integrasi ini dilakukan adalah model-model simulasi boleh dibandingkan dengan mudah apabila input-input parameter yang berbeza digunakan. Ini kerana hasil keputusan untuk setiap model tersebut boleh disimpan di dalam storan data dan digunakan semula pada bila-bila masa. Projek yang dilakukan ini memberi pengkhususan kepada pendekatan yang boleh diambil untuk mengintegrasikan *Web-based simulation* dengan storan data. *Four-tiered architecture* adalah bentuk senibina yang dicadangkan untuk projek ini, dan beberapa teknik diambil kira dalam membina konsep tersebut, selain daripada ciri-ciri keselamatan yang boleh diambil.

Katakunci : simulasi, *Web-based simulation,* storan data, *tiered-architecture*

ABSTRACT

Vast and rapid technological innovations and advancement have created an evolution to the field of simulation. The birth of the Internet and the World Wide Web (WWW) has significantly influenced many aspects of conventional simulation techniques. In addition, the Web technologies have the potentials to alter the ways in which simulation models are developed, documented, analyzed and executed. The deployment of the Web technologies in the simulation application has given a new birth to the field, a concept known as Web-based simulation. One of the advantages that can be taken from the Web advent technologies is the integration of the Web-based simulation with databases. The most prominent advantage is simulation models using different input parameters can be easily compared as the results will be stored in the database server, and can also be retrieved anytime. In addition, it also enables information sharing across different groups for various decisions making based on the models created. The study conducted focuses on approaches that can be taken to integrate Web-based simulation with databases. A four-tier architecture is proposed as a resolution to the limitations of two and three-tier architecture. Several tools and techniques are taken into consideration in developing a conceptual framework for the environment in addition to security features that should be incorporated to ensure a secure link to the Internet and Web users.

Keywords : simulation, Web-based simulation, database, tiered-architecture.

ACKNOWLEDGEMENT

In the name of Allah, the Most Gracious and the Most Merciful.

I would like to thank :

My beloved parents for their fullest supports, love and encouragement,

Universiti Teknologi Mara and Universiti Utara Malaysia for the facilities and resources provided,

My supervisor, Prof. Madya Dr. Razman Mat Tahar for his help, and constructive comments and ideas,

All UUM lecturers who have thought me in the MSc. IT program,

My sisters, brother, in-laws and friends for their understanding and motivation,

And all the individuals involved in the establishment of this project.

TABLE OF CONTENTS

ABSTRAK i
ABSTRACT ii
ACKNOWLEDGEMENT iii
TABLE OF CONTENTS iv
LIST OF FIGURES vi
LIST OF TABLES vii

CHAPTER 1 : INTRODUCTION 1
1.1 Problem Statement 2
1.2 Objective of Research 4
1.3 Scope of Research 5
1.4 Methodology 6
1.5 Thesis Outline 6
1.6 Significance of Research 7

CHAPTER 2 : INTERNETWORKING AND WORLD WIDE WEB (WWW) 8
2.1 Web Components 9
2.1.1 The Internet 10
2.1.2 Information Servers 11
2.1.3 Web Browsers 11
2.2 Web Fundamentals 12
2.2.1 The HyperText Markup Language (HTML) 12
2.2.2 The HyperText Transfer Protocol (HTTP) 12
2.2.3 Uniform Resource Locators (URL) 13
2.2.4 Common Gateway Interface Pograms (CGI) 15
2.2.5 Java Applets 16
2.2.6 ActiveX – Microsoft Objects 17
2.3 The Web As A Database Application Platform 18
2.3.1 Two-tier Architecture 20
2.3.2 Three-tier Architecture 21
2.4 Database Connectivity Solutions For The Internet 23
2.4.1 Common Gateway Interface (CGI) 24
2.4.2 Middleware Solutions 25
2.4.3 Distributed Object Frameworks 26
2.4.4 Java and JDBC 27
2.5 Establishing Network Security 28
2.5.1 Web Security Threats 29
2.5.2 Web Security Countermeasures 30

CHAPTER 3 : SIMULATION AND THE EMERGENCE OF WEB-BASED SIMULATION 34
3.1 Simulation Overview 35
3.1.1 Purposes of Simulation 36
3.2 Steps in a Simulation Study 37
3.3 Areas and Applications of Simulation 41
3.4 Web-Based Simulation Overview 42
3.4.1 Comparisons of Web-based Simulation to Other Architectures 43
3.5 Supported Steps in a WBSE 44
3.5.1 How Web Technologies Can Support Simulation 44
3.6 WBSE Concept and Components 46
3.7 Approaches To Integrating WBSE and Databases 47
3.8 WBSE and Security Concern 48

CHAPTER 4 : ESTABLISHING A CONCEPTUAL FRAMEWORK FOR A FOUR-TIER WEB-BASED SIMULATION ENVIRONMENT 51
4.1 Rapid Application Development 51
4.2 Approaches to Designing the Framework 53
4.3 Designing a Conceptual Framework for a 4-Tier Web-based Simulation Environment 58
4.3.1 Baseline 1 60
4.3.1.1 Graphics Design 60
4.3.1.2 System Interfaces 61
4.3.2 Baseline 2 66
4.3.2.1 Network and Web Tools Integration 66
4.4 Towards the Implementation 69
4.4.1 Java Database Connectivity 70
4.5 Security Tools Integration 73

CHAPTER 5 : CONCLUSION 75

REFERENCES 77

GLOSSARY 83

LIST OF FIGURES

Figure 1- 1 : Basic Proposed WBSE Architecture 4
Figure 2- 1 : Basic Architecture of the World Wide Web 10
Figure 2- 2 : Java Database Access via Internet/Intranet 17
Figure 2- 3 : Layers of Database Application 20
Figure 2- 4 : Two-tier Client-Server Architecture 21
Figure 2- 5 : Three-tier Network Architecture 22
Figure 3- 1 : Steps in a Simulation Study 40
Figure 4- 1 : Traditional Development vs. RAD 52
Figure 4- 2 : RAD Four Phases 53
Figure 4- 3 : System Architecture of the Prototype 56
Figure 4- 4 : Sequence Diagram for a Typical Startup Scenario 57
Figure 4- 5 : A General Layout of the 4-Tier Web-based Simulation Environment 60
Figure 4- 6 : Enhanced Model of the Proposed WBSE 62
Figure 4- 7 : CORBA Infrastructure in the WBSE 65
Figure 4- 8 : A Detailed Architecture of the WBSE 66
Figure 4- 9 : Relationships Among Classes 68
Figure 4- 10 : The JDBC API's Tasks 70
Figure 4- 11 : Block Diagram of Database Application 71

LIST OF TABLES

Table 2- 1 : Web Security Threats.....30
Table 3- 1 : Some Typical Applications of Simulation.....42
Table 3- 2 : Characteristics Comparison of Simulation Architectures.....43
Table 4- 1 : System Life-cycles.....59
Table 4- 2 : Object Relationship.....67

CHAPTER 1 : INTRODUCTION

The emergence of the World Wide Web (WWW) has produced very large impacts to many disciplines in which their approaches, techniques and philosophies have to be re-evaluated (Page and Opper, 1999). The WWW is the most powerful and popular portion of the Internet, and since its first publicly available in 1993, many organizations and individuals have taken full advantages of its potentials. Computer simulation field is not an exception to this phenomenon. In a simple explanation, a simulation is defined as an imitation of a real world process or system (Banks, 1999). A more precise definition of simulation as given by Shannon is ; "simulation is the process of designing a model of a real system and conducting experiments with this model for the purpose of either understanding the behavior of the system and/or evaluating various strategies for the operation of the system" (1992). Generally, simulation is used as a methodology to describe and analyze the behavior of a system, where the 'what if' questions are the keys to aid in designing the real system (Banks, 1999).

Apart from engaging and complying to various users' and clients' needs that keep changing, those involve in this field have seriously begun to take the advantages of enabled advanced technologies and exploit the benefits of the WWW for promoting and improving simulation techniques. The attendant technologies of the WWW have driven a new theme in simulation research and practice, a concept known as web-based simulation.

In a simplest term, a web-based simulation is a method of engaging the typical simulation steps via the available WWW technologies. The major advantages of web-based models are platform independence, reusability and interoperability (Ook, 1999), which are the important elements for an efficient network operation.

1.1 Problem Statement

Web-based simulation has only emerged in the mid 1990s and Fishwick, being one of the pioneers, describes several potential impacts of web-technologies on simulation. Specifically, these areas are 1) education and training, 2) publications and 3) simulation programs (Page & Opper, 1999). Web-based simulation has given tremendous benefits to these fields due its independence, reusability and interoperability. Other benefits include the scalability for multiple users, ability to facilitate distance-group decision making, ability to facilitate on-line training and intuitive to use, reusability to design for simulation models, ability to use multimedia data resources and platform-free simulation modeling (Ook, 1999).

One of the possible architecture of a web-based simulation is its integration with DBMS. This architecture can provide tools for sending input data to simulation models and collecting output results from the models through the web (Ook, 1999). Although the linkage between the WBSE and the database offers several important strengths, it also has few weaknesses. For instance, the two-tier architecture that links the client to the web server has its limitation, where it is not suitable for the existing simulation and animation tools. On the other hand, the three-tier architecture that links a Web-client, a simulator and a database server does not allow the user to interrupt a running simulation because the web server send the results only at completion (Ook, 1999). (Details of two and three-tier architecture are discussed in Chapter 2). Hence, a four-tier architecture can be designed to overcome the shortages of the architectures mentioned. However, previous research on WBSE conducted for instance by Iazeolla and D'Ambrogio (1998) does not have a connection with a database, and the main focus of the study is CORBA-based architecture. Thus, this research will be conducted to design a conceptual four-tier architecture of a web-based simulation environment that integrates with database technology. Apart from that, common network architecture components such security will also be highlighted since these fundamental elements are usually left out in most of the WBSE design literature.

1.2 Objective of Research

The main objective of the research is to design a conceptual framework of a four-tier architecture for a web-based simulation environment that integrates with database technology. Basically, this architecture consists of the web client as the 1st tier, the web server as the 2nd tier, the simulator as the 3rd tier and a database server as the 4th tier. Several possible techniques and tools will be identified to allow simulation users to store results in an external database server, and design the results with a personal preference for easy understanding. In addition, several essential network elements will also be included in the study. The following diagram depicts the proposed architecture of the research.

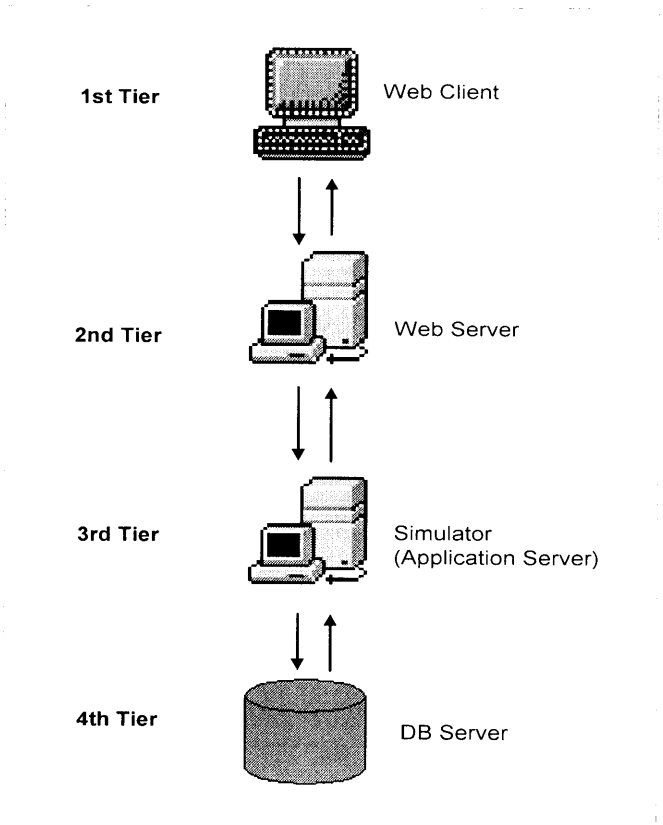

Figure 1 - 1 : Basic Proposed WBSE Architecture

1.3 Scope of Research

One of the key elements of an application design is the system architecture. The system architecture defines interaction between different pieces of the application and the functions that each piece is responsible for. In any web-based environment, applications that are built exist in a distributed environment and a distributed application separates the application functionality into more manageable group of tasks. Ultimately, this creates a series of application layers or tiers, where each tier is responsible for an individual element of the application's processing. In designing the architecture for the study, several potential issues will be identified, which include:

- To determine the appropriate way to structure the overall application architecture
- To determine the best way to group the objects together
- To determine the most appropriate tools and techniques that can be used for the environment
- To study the strengths and benefits of a four-tier architecture of a WBSE
- To provide a secure link to the Internet and WWW users

In addition, the research will focus on the tools and techniques needed to design the four-tier architecture. For instance, using Java Servlet, JDBC Driver and pure distributed objects for better deployment. However, since the study gives a close focus on designing a conceptual framework of a Web-based simulation environment, it will only be conducted up to the design phase using Rapid Application Design

methodology. Due to several limitations, the proposed architecture will not be constructed physically for the research.

1.4 Methodology

The main methodology used for the study is Rapid Application development (RAD). RAD is chosen since it allows for lifecycle iteration, which contributes to a more effective development. However, due to some limitations, only two phases of the methodology are applied, and the study is conducted up to the design stage. Apart from that, an informal interview has been conducted with an officer from *Pusat Komputer, Universiti Utara Malaysia* to get some useful commentary on the work. Other methods used to gather information include library research and informal discussions.

1.5 Thesis Outline

The documentation is made up of five chapters as outlined below :

- Chapter 2 : provides a general introduction of the Internet and the World Wide Web (WWW). It gives an overview of the fundamental components of the Web and the techniques and technologies that form the Web aspects. Various Internet-database connectivity techniques are also outlined, in addition to Internet security threats and countermeasures that can be taken.
- Chapter 3 : gives a general outline of simulation and the key factors that drive the conventional technique to a new form of application, a concept known as Web-based-simulation. It also gives the overview of Web-based simulation

components and some related work conducted by others. Apart form that, this chapter also introduces the approaches to Web-based simulation and database integration, and the security issues that have to be concerned.

- Chapter 4 : is concerned with the designing of four-tier Web-based simulation conceptual framework. It discusses the methodologies used in developing the framework, the tools and techniques applied for the system integration and the security features that can be added to the design.
- Chapter 5 : describes the evaluation of the study that leads to the future enhancements.

1.6 Significance of Research

It is hoped this little research will give more understanding of Web-based simulation application and its associated tools and techniques. Hence, more attempts can be made to apply the concept in greater areas. The adoption of simulation in general and Web-based simulation in particular is quite limited in Malaysia, although these applications have proven to be very useful in problem solving and decision making. Therefore, rationale steps have to be taken to enhance the application via the advent technologies available.

CHAPTER 2 : INTERNETWORKING AND WORLD WIDE WEB (WWW)

The birth of the Internet has created a major change in the history of communication and computerisation. The Internet is the world's largest computer network (Cheng and Malaika, 1996), which is a mechanism for information dissemination and a medium for collaboration and interaction between individuals and their computers without regard for geographic location (Leiner et al, 2000). Historically, the Internet began as the United States government's Department of Defence designed a secure and survivable communications network for organisations engaged in defence-related research in 1969. This project was known as ARPANET (Advanced Research Projects Agency Network) and created by the Pentagon's Advanced Research Projects Agency (wdvl.com, 2000). In 1970's, the ARPANET grew bigger and it was upgraded to a high-speed network by linking several powerful supercomputer stations called nodes. In the 1990's, the Internet has grown more rapidly and today, Internet's influence reaches not only the technical fields of computer communications, but throughout the society as a whole as "people are moving toward increasing tools to accomplish electronic commerce, information acquisition and community operations" (Leiner et al, 2000).

One of the most prevalent services provided via the Internet is the World Wide Web or simply the Web. It combines the endeavour of exploring new and stimulating destinations, with the excitement of multimedia such as playing a video game, listening to music or directing one own movie. Briefly, the Web is the "collection of browsers, servers, files, and browser-accessible services available trough the Internet" (Perl, CGI and JavaScript, 2000). It was created by Tim Berners-Lee, who worked at the *Conseil Europĕen pour la Recherché Nucléaire* (CERN), located at Geneva Switzerland in 1989. The original purpose for the Web design was to facilitate communication between research scientists. Today, the Web has become popular primarily due to its simplicity where users can easily access and deliver objects anywhere via the Internet. The Web also provides vast amount of potentials and functionality for various types of usage. The potentials offered are the main reason for the Web to surge high popularity in any industry and organisation (Teleen, 1996). The following section will discuss the essential components that together form the World Wide Web and the major technologies used in the environment.

2.1 Web Components

There are three major components that form a basic structure of the Web. These components are the *Internet*, the *Information Servers* that contain and disperse information, and the *Web browser* whom the individuals use to obtain information and pages from the web (webcom.com). *Web browsers* use the *Internet* to access

servers that contain pages, images and other files that users interested in receiving. The following diagram illustrates a basic web architecture.

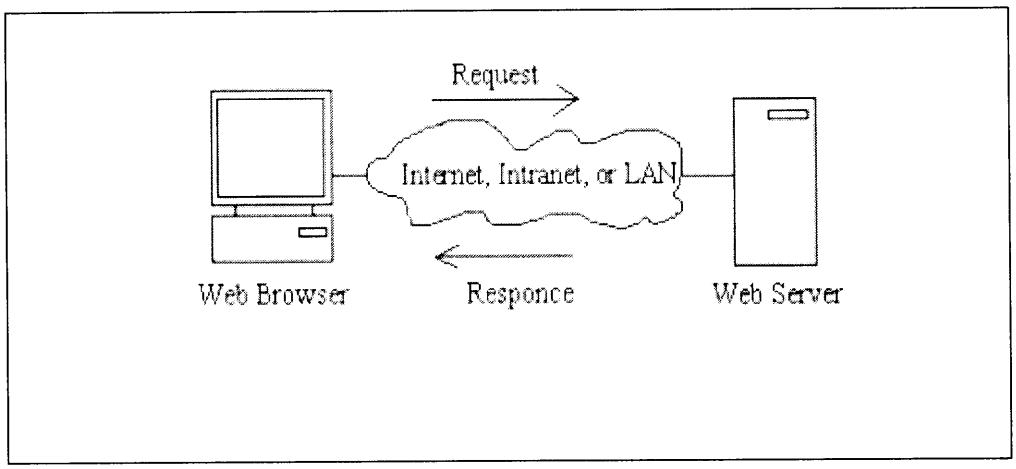

Figure 2- 1 : Basic Architecture of the World Wide Web

2.1.1 The Internet

On the technical level, the Internet consists of wires, cables, machines and networking software that connect millions of computers around the world (webcom.com). On the other hand, by looking at the network components, Yarborough has outlined several enabling technologies and contents that form the network objects of the Internet (1998).

Since the Web uses the Internet as its communication medium, it must adhere to the Internet rules and procedures for exchanging information, known as protocol. The Internet's Transmission Control Protocol (TCP) and Internet Protocol (IP) enable worldwide connectivity between browsers and servers. In addition to TCP/IP, the Web also uses its own protocol, called the Hyper Text Transfer Protocol (HTTP) for exchanges between browsers and servers (Perl, CGI and JavaScript, 2000).

2.1.2 Information Servers

Information servers are executing computer software that bring out information requested by users. The servers also provide gateways that enable browsers to access Web-related applications, such as database searches and electronic payment systems (Perl, CGI and JavaScript, 2000). There are various types of information servers on the Internet, and some of them are WWW servers, which are also called http servers and they primarily deliver data for immediate usage; gopher servers which present files in distributed archives; file transfer protocol (FTP) servers which allow FTP clients to copy files between client and server machines; Wide Area Information Servers (WAIS) which search distributed volumes of text and Network News Transport Protocol Servers (NNTP) which deliver Usenet newsgroups and articles (webcom.com).

2.1.3 Web Browsers

The third component of the Web is web browsers, which is the most familiar element of the web to many people. Basically, the browser is the user's window to the Web systems (Perl, CGI and JavaScript, 2000), and it is a program for interfacing to a web server (Birman, 1996). The most popular browsers nowadays are Netscape Navigator and Microsoft's Internet Explorer, which are descendants of the Mosaic browser. Today's browsers extend Mosaic's features with multimedia capabilities and with browser programming languages such as Java and JavaScript to make browsing more efficient and exciting.

2.2 Web Fundamentals

There are various techniques and technologies that form the fundamental elements of the Web and this section discusses some of these basic aspects.

2.2.1 The HyperText Markup Language (HTML)

The HyperText Markup Language, or the HTML is "a standard for representing textual documents" (Birman, 1996) or in other words, it is used to publish documents on the Web. In simplest form, it is the lingua franca of the Web, and is used to create web pages. Somehow, HTML is similar to the codes used by some word processing programs, for instance WordPerfect (Perl, CGI and JavaScript, 2000).

An HTML provides two fundamental functions; hyperlink to refer or link to other documents, and hypermedia to include multimedia forms such as images, sound, animation and video (Cheng and Malaika, 1996).

2.2.2 The HyperText Transfer Protocol (HTTP)

According to Cheng and Malaika, HTTP, which is a generic, stateless, object-oriented protocol, is an application-level protocol for retrieving documents with a fast response time over the Internet (1996). HTTP uses a request/response model of communication between browsers and web servers and it provides an efficient method of finding resources using URL (www.esatclear, 2000). The process begins when a browser establishes a connection with a server and sends URL requests. The requests specify the request method, the URL of the document, program, HTTP version being used by the browser and other related information. The server then

processes the requests and sends a response back to the browser. Among the most common HTTP methods issued from browsers to servers include (Cheng and Malaika, 1996 ; Perl, CGI and JavaScript, 2000):

- the GET method is used to retrieve information and objects from Web servers at the specified URL. This method may also be used to submit data collected in an HTML form or to invoke Common Gateway Interface program.
- The HEAD method, which is similar to the GET method, except it is used to retrieve information about a URL without actually obtaining the information addressed by the URL.
- The POST method is used to send user input, for instance form data and other information to Common Gateway Interface programs. The Web server responds to a POST request by sending back header data followed by an information generated by the CGI programs that results from the request.

On the Internet, HTTP communication generally takes place over TCP/IP connections. It uses as default port 80, but other ports can be used, which does not prevent HTTP from being implemented on top of any other protocol (Cisco Networking Academy Program, 2000)

2.2.3 Uniform Resource Locators (URL)

A Uniform Resource Locator, or URL "is the notation used to specify the addresses of an Internet file or service" (Perl, CGI and JavaScript, 2000). It indicates "which

Internet service is used, where to send the request and what the request is (Cheng and Malaika, 1996). A URL always contain the following format:

protocol identifier://*hostname*[:*domain_port*] *path*[/*request*]

where

- *protocol identifier* can be one of the following:

file for local file
ftp for file transfer
gopher for gopher service
http for hypertext documents
mailto for outgoing mail
news or nttp for usenet news
telnet for connecting to other computers
wais for wais database search

- *hostname* is the name of the host computer
- *domain_port* is the network port number, and it is optional. The default is 80 for http; 70 for gopher
- *path* is the path to get to the request
- *request* can be a filename.file_extension or a query.

On the other hand, the main functions of the URL are as the following (Yarborough, 1998):

- to allow WWW browser clients to indicate the target of the hyper-text links
- to specify Internet resources
- to provide the Internet domain name of the site on which the server is running
- to supply the port number

- to furnish the location of the resource in the hierarchy structure of the server
- to create the string argument for an HTTP URL.

2.2.4 Common Gateway Interface Pograms (CGI)

The Common Gateway Interface (CGI) is "a standard that specifies how external programs may be used by Web servers" (Perl, CGI and JavaScript, 2000). According to Cheng and Malaika, CGI defines an interface between a Web server and a program (1996) and the main function of the CGI program is to process the input parameters and produce the output as an HTML page. Although CGI performance can be considered slow for complex executions, it is fast enough for many simple applications. The following are some of the common uses for the CGI (Cheng and Malaika, 1996):

- Visitor page count : to display the number of times a page has been visited within the page itself
- Server-side includes (SSI): to enable a Web server to insert variable items into static and dynamic Web pages at runtime. An SSI is particularly beneficial for boilerplate information such as copyright notices and last date of page modification
- Clickable image maps : to allow another URL to be displayed when a user clicks on a portion of a displayed image
- Gateways to existing systems: to enable a Web browser to excess an existing system, and a server program is used to do a translation between the CGI and the interface language, such as SQL.

More information on CGI will be discussed in later section of this chapter.

2.2.5 Java Applets

Java applets are Java programs that can be executed by the Web browser. They are called applets rather that applications since they cannot be run outside of the browser's own window (Perl, CGI and JavaScript, 2000). When a web page that references a Java applet is loaded by a browser, the browser requests the applet code from the Web server, and it will be downloaded and executed within the client's Web browser (www.esatclear, 2000). However, Java applets have many strict restrictions placed on them. The first major restriction is they have limited actions to be discharged to ensure they do not perform any harmful operations on the client machine, in addition to restriction to modify existing data contained on the client. Secondly, applets can open connections back to the server from which they were originally downloaded. However, if the source of the originating server is certified as being trustworthy, these restrictions can be discarded.

Java applets also enables database connectivity for Internets and Intranets via Java Database Connectivity (JDBC). JDBC makes a Java program possible to connect to and perform database operations on any database either locally or across a network, for which the correct drivers exist. The following figure shows how database is accessed via the Internet using Java applets.

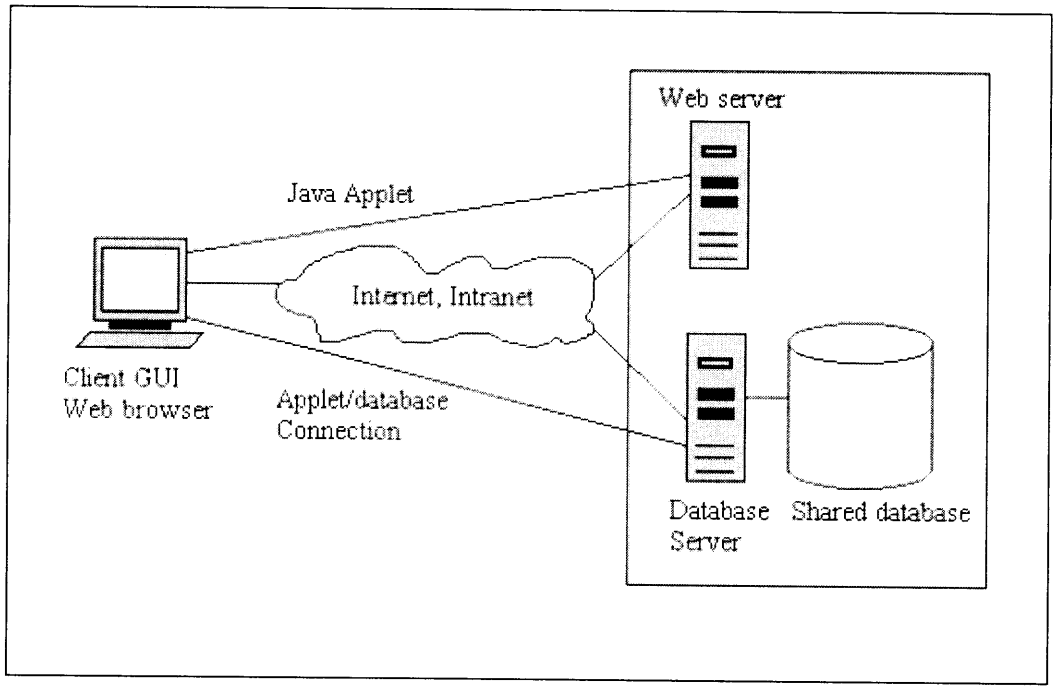

Figure 2- 2 : Java Database Access via Internet/Intranet

2.2.6 ActiveX – Microsoft Objects

ActiveX is Microsoft's approach to executing objects other than Java applets in Internet Explorer and is gaining its popularity (www.esatclear, 2000). Although the name might seem like a new and innovative technology, it is nothing more than a Component Object Model (COM) objects that can be downloaded and executed by Internet Explorer (Perl, CGI and JavaScript, 2000). Across the Internet, ActiveX controls are stored on a Web server and can be downloaded on request like a Java applet.

ActiveX can be used to connect database for Internets and Intranets via Microsoft's Open Database Connectivity (ODBC). Similar to Java JDBC, ODBC provides a low-level API. However, due to the fact that ActiveX security model is currently not

effective as the Java security model, it is not ideal for database interfaces across the Internet (JC, 1998).

2.3 The Web As A Database Application Platform

The history of database system research is one of exceptional productivity and startling economic impact. Barely 20 years old as a basic science research field, database research has fueled an information service industry estimated at $10 billion per year in the U.S alone. Achievements in database research underpin fundamental advances in communications systems, transportation and logistics, financial management, knowledge-based systems, accessibility to scientific literature, and a host of other civilian and defense applications. They also serve as the foundation for considerable progress in the basic science fields ranging from computing to biology.

(Silberschatz et al., 1991)

The above excerpt is quoted from a book entitled "Database Systems: A Practical Approach to Design, Implementation and Management" by Connolly and Begg, and from the quotation, they argue database system is the most important development in the field of software engineering (1999). In today's technologically advanced society where information plays a vital role, having good information storage location from which data can be retrieved and manipulated is really critical. Hence, databases play an important role in the process of gathering, processing and distributing information.

As more and more businesses and organizations are taking full advantages of the Web potentials, storing and retrieving the relevant data from the Web is one of the major criteria that has to be highly and carefully thought about. This is due to the fact

that the Web provides access to information on the global scale (www.esatclear, 2000), and it allows the contents of a database residing in the Web environment to be shared worldwide. Connolly and Begg have highlighted some of the most important requirements for the integration of the database applications with the Web (1999). These include:

- The ability to access valuable corporate data in a secure manner
- Data and vendor independent connectivity to allow freedom of choice in the selection of the Database Management System (DBMS) now and in the future
- An open-architecture approach to allow interoperability with a variety of systems and technologies. For instance, support for :
- Different Web servers;
- Microsoft's (Distributed) Common Object Model (DCOM/COM);
- CORBA/IIOP (Internet Inter-ORB protocol);
- Java/Remote Method Invocation.
- Support for transactions that span multiple HTTP requests
- Acceptable performance and minimal administrative overhead
- A set of high-level productivity tools to allow applications to be developed, maintained and deployed with relative ease and speed.

According to Dataramp, network database applications can be divided into three parts, which are the user interface, the business logic and shared data source (www.esatclear, 2000), as shown in the next figure. Generally, the business logic contains the rules specifically designed for database application, and it is where all

the necessary code to process user requests resides. In addition, it performs functions to validate, transform and request data. Both the user interface and the business logic make up the application logic. The shared data source is normally in the form of a database management system, which is responsible for managing the data and enabling it to be updated, shared and retrieve.

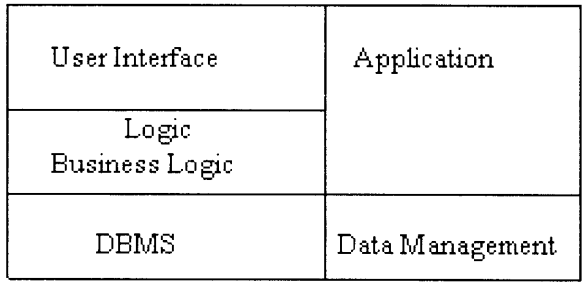

Figure 2- 3 : Layers of Database Application

When a database application is deployed over a network, for instance the Internet or LAN, it must be partitioned into two or more parts or tiers (Connolly and Begg, 1999). Orfali suggests there are many variations of multi-tier architectures depending on the application and middleware used to establish communications between the www.esatclear, 2000). The most common architectures are the two-tier architecture and the three-tier architecture.

2.3.1 Two-tier Architecture

The traditional two-tier client-server architecture provides a basic separation of tasks from the conventional mainframe environment. Two-tier applications usually involve thick or fat clients, those which contain processing rules as well as presentation logic (Yarborough, 1998). In this architecture, the client (tier 1) is primarily responsible

for the presentation of data to the user whole the server (tier 2) is primarily responsible for supplying data services to the client (Connolly and Begg, 1999), as illustrated in the following figure. However, two-tier implementation may face scalability, performance and flexibility problems (www.esatclear, 2000), and larger systems usually adopt a three-tier or multi-tier approach for better performance.

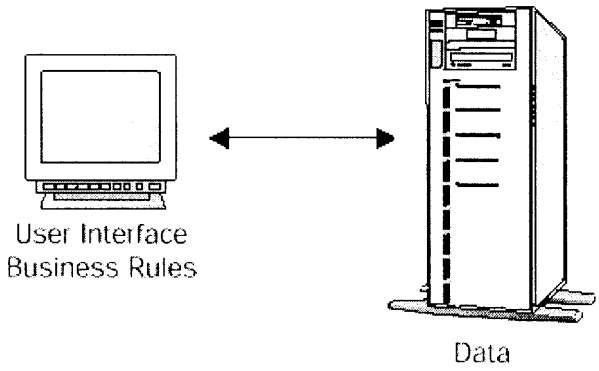

Figure 2- 4 : Two-tier Client-Server Architecture

2.3.2 Three-tier Architecture

The two-tiered network architecture drawbacks and the need for enterprise scalability have resulted in the adoption of a three-tiered or multi-tiered architecture. This approach uses an application server between the clients and the back-end database server, as shown in the next diagram.

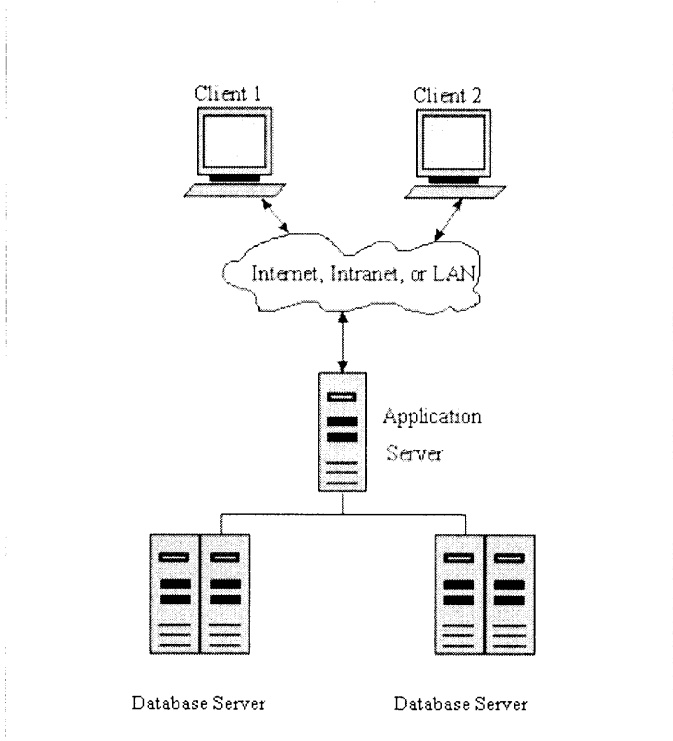

Figure 2- 5 : Three-tier Network Architecture

Each layer in the architecture runs on a different platform as the following (Connolly and Begg, 1999) :

1. the user interface layer, which runs on the end-user's computer (the client)
2. the business logic and data processing layer (middle tier), which runs on a server and is often called the application server
3. a DBMS, which stores the data required by the middle tier. This tier may run on a separate server known as database server.

The three-tier architecture has many advantages over the traditional two-tier design, which include (Connolly and Begg, 1999):

- the 'thin' client, which requires less expensive hardware
- centralized application maintenance by centralizing the business logic into a single application server. Thus, it eliminates software distribution limitation
- the added modularity makes it easier to modify or replace one tier without affecting the other tiers
- separation of core business logic from the database functions makes it easier to implement load balancing.

In addition, there other potential benefits of three-tier or multi-tier architecture, which among others include more scalable and easier to control, better security and easier maintenance, simplified connectivity to heterogeneous data as the required database drivers are contained at a single location, which then require fewer client licenses (www.esatclear, 2000).

2.4 Database Connectivity Solutions For The Internet

There are several ways Web based applications can be connected to databases.

Among the common approaches are:

1. CGI
2. Middleware solutions
3. Universal and Web aware databases
4. Distributed object frameworks
5. Java and JDBC

2.4.1 Common Gateway Interface (CGI)

CGI is a specification for transferring information between a Web server and a CGI program (Connolly and Begg, 1999). According to Bottjer (1997), the CGI Application architecture was the first model developed to extend the capabilities of the Web to include active resources, and still one of the most flexibles. The main reason for CGI being the main model for deploying applications across the Internet is it is supported by virtually every Web server and browser, making it possible to bring platform independent interactive applications to anyone with a Web browser (www.esatclear, 2000). The following steps are carried out when a user requests information from a database via the Internet (www.esatclear, 2000 ; Connolly and Begg, 1999 ; Orfali et al, 1999):

1. The user calls the CGI script by clicking on a link or by pushing a button
2. The Web browser requests the CGI script link and transmit the user input to the Web server
3. The server checks the configuration and access files to ensure the users has access to the CGI script and check that the CGI script exists
4. The server prepares the environment variables and launches the script
5. The script executes and reads the environment variables and STDIN
6. The script sends the proper MIME headers to STDOUT followed by the remainder of the output and terminates
7. The server sends the data in STDOUT to the browser and closes the connection
8. The browser displays the information sent from the server.

CGI offers some advantages, which include simplicity, language independence, Web server independence and wide acceptance (Connolly and Begg, 1999). However, despite its benefits, there are several drawbacks associated with CGI–based approach:

- Lack of speed due to the reliance of HTTP, and this makes CGI/HTTP unsuitable for developing many critical applications.
- Lack of efficiency and transaction support, which essentially is inherited from the statelessness of the HTTP protocol
- All the processing occurs on the server side, requiring a round trip to provide the user with feedback or input validation

2.4.2 Middleware Solutions

Middleware, which is the enabling technology behind client/server computing (Tristram, 1996), plays an important role in developing client/server applications (Freeman, 1997). Middleware can be divided into categories, which according to Hurwitz are based on scalability and recoverability (McFadden et al, 1999):

1. Asynchronous Remote Procedure Call (RPC): it has high scalability but low recoverability. The client usually establishes a point-to-point connection with the server and performs other processing while it waits for the response.
2. Publish/subscribe: monitors activity and pushes information to subscribers. This type of middleware is very useful for monitoring situations where actions need to be taken when particular events occur.

3. Message Oriented Middleware (MOM): similar to asynchronous RPC, MOM sends message that are collected and stored until they are acted upon, while the client continues with other processing.
4. Object Request Broker: enables applications to send objects and requests services in an object-oriented system. ORB removes complex RPCs by providing the mechanisms by which objects make and receive requests and responses transparently (Connolly and Begg, 1999).
5. SQL-oriented data access: translates generic SQL into the SQL specific to the database.
6. Transaction Processing (TP) monitor: provides an execution environment for server based mission critical applications which run repetitively in production (Cobb et al, 1996).

According to Tristram, although middleware is very important and bring a lot of benefits to tiered architecture, it is quite often proprietary, difficult to write and costly (www.esatclear, 2000). However, Web technologies can be used as a cheap and easy solution due to its open standards.

2.4.3 Distributed Object Frameworks

Distributed objects are another solution for connecting heterogeneous databases across different platform. Similar to classical objects, distributed objects inherit attributes and actions, and call each other's method, but the latter can be physically separated and located anywhere across a network (Öszu and Valduriez, 1999).

According to Pope, the concept of distributed objects comes from the combination of two paradigms – client server and object oriented, with some explicit differences (1998):

- a client knows an object by its interface
- objects are not always local with respect to their clients
- dynamic composition may compose objects into new applications
- objects hide many of the underlying differences in architecture through encapsulation.

Currently, there are two main types of distributed objects available. The main standard by the Object Management Group is the Common Object Request Broker (CORBA), while the second standard is Distributed Object Component Model (DCOM), developed by Microsoft. Comparing the two, CORBA which is the older and more widely used, is slightly faster than DCOM and easier to implement and more reliable (www.esatclear, 2000).

2.4.4 Java and JDBC

JDBC technology is an Application Programming Interface (API) that let users to access tabular data source from Java programming language. It provides cross-DBMS connectivity to a wide range of SQL databases (javasoft.com, 2000). The JDBC API allows developers to take advantage of the Java platform's "Write Once, Run Anywhere™" capabilities for industrial strength, cross-platform applications that require access to enterprise data. With a JDBC technology-enabled driver, a

developer can easily connect all corporate data even in a heterogeneous environment. More details on JDBC will be discussed in the later section of this document.

2.5 Establishing Network Security

Security can be defined as providing information access to the appropriate personnel, while at the same time barring access to others (Randy, 1996). In addition, there are three (3) basic concepts of security, which are privacy (confidentiality), authentication and data integrity. In addition, according to Connolly and Begg (1999), the challenge of the Internet is to transmit and receive information safely while ensuring:

- It is inaccessible to anyone but sender and receiver (privacy)
- It has not been changed during transmission (integrity)
- The receiver can be sure it came from the sender (authenticity)
- The sender can be sure the receiver is genuine (non-fabrication)
- The sender cannot deny he or she sent it (non-repudiation)

There are common possible attacks and threats that adversely affect a system, and consequently an organization. Some of these as outlined by Dodds et al are (2000):

- Spoofing User Identity : a hacker is being able to impersonate a valid system user or resource to get access to the system, hence compromise system security
- Tampering with Data : a malicious intruder makes changes to stored or in-transit information, introduces an undetectable network packet in a communication and makes undetectable changes to a sensitive file
- Repudiability : a user performs an illegal operation without the ability to be traced

- Information Disclosure : a user is able to read a file that he/she is not granted access to, and he/she is also able to read the data while in transit
- Denial of Service (DoS): prevents legitimate users from using a service. The effectiveness of a DoS attack is measured in three ways, which are effort, severity and persistence
- Elevation of Privilege: an unprivileged user gains privileged access, thus has sufficient access to compromise or destroy the entire system.

2.5.1 Web Security Threats

The security tools and approaches of a client/server application are relevant to the issue of Web security. However, the Web presents new challenges in the context of computer and network security due to the complexity of data, platform independent, complex software and common users who are not aware of security risks and countermeasures. The following table summarizes a comparison of Web security threat and countermeasures that can be taken.

	Threats	Consequences	Countermeasures
Integrity	• Modification of user data Trojan horse browser • Modification of memory • Modification of message traffic in transit	• Loss of information • Compromise of machine • Vulnerability to all other threats	Cryptographic checksums
Confidentiality	• Eavesdropping on the Net • Theft of info from server • Theft of data from client • Info about network configuration • Info about which client talks to server	• Loss of information • Loss of privacy	Encryption, Web proxies
Denial of Service	• Killing of user threads • Flooding machine with bogus threats • Filling up disk or memory • Isolating machine by DNS attacks	• Disruptive • Annoying • Prevent user from getting work done	Difficult to prevent
Authentication	• Impersonation of legitimate users • Data forgery	• Misrepresentation of user • Belief that false information is valid	Cryptographic techniques

Table 2- 1 : Web Security Threats

2.5.2 Web Security Countermeasures

The increasing accessibility of databases over the Internet requires thorough analyses and approaches associated with security. Several countermeasures can be taken and some of the common approaches are listed below: (Connolly and Begg, 1999 ; Bottjer, 1997; Cheng and Malaika, 1996 ; Yarborough, 1998 ; Pope, 1998 ; Bates, 1998 ; Beyda, 1996):

1. Proxy servers

In a Web environment, a proxy server is a computer that sits between a Web browser and Web server. Conceptually, a proxy server acts as a liaison between the application and the external network. However, in reality, the proxy server creates

another link to the outside server, and exchange the information on behalf of the Web browser. It serves two main purposes: to improve performance and filter requests. By acting as a liaison, the proxy server controls the traffic that flows through it and saves the results of users' requests; thus it significantly improves the performance. In addition, it can also be used to filter requests to prevent unauthorized users from access a specific set of Web sites.

2. Firewalls

A firewall is simply a 'barrier' that sits between an internal network and an external network, with the primary objective is to prevent unauthorized access to and from a private network. Based on a set of predefined rules, a firewall determines whether to pass data through or to discard it. Firewalls are usually not separate devices, but rather are functions performed inside existing network equipment. There are several types of firewall techniques:

- Packet filter: accepts or rejects packet entering or leaving the network based on user-defined rules, and it is susceptible to IP spoofing.
- Applications gateway: applies security mechanisms to a specific application, for instances FTP and Telnet servers.
- Circuit-level gateway: applies security mechanisms when a TCP or User Datagram Protocol (UDP) connection is established.
- Proxy server: intercepts all messages entering and leaving the network.

3. Message Digest Algorithms and Digital Signature

A message digest algorithm, or one way hash function, takes an arbitrarily-sized string, which is the message, and generates a fixed-length string, which is the digest or hash. A digest should be computationally infeasible to find another message and it does not reveal anything about the message. On the other hand, a digital signature is a digital fingerprint that consists of two pieces of information: data that has been 'signed' along with the private key of the individual and organizations wishing the signature. Like a handwritten signature, the digital signature cannot be forged and the signed data cannot be changed.

4. Digital Certificates

A digital certificate is an attachment to an electronic message to verify the sender is the actual person he/she claims to be. The certificate is also used to provide the user with the means to encode a reply. A digital certificate can be applied from a Certificate Authority (CA) and the most widely used standard for digital certificates is X.509.

5. Kerberos

Kerberos, which is named after the three-headed monster in Greek mythology that guarded the gate of hell, is a server of secured user names and passwords. The main function of Keberos is it provides one centralized security server, which features database access, login and authorization control for all data and resources on the network. Kerberos also has similar function to that of a Certificate server: to identify

and validate a user. Currently, security companies are taking some actions to merge Kerberos and Certificate servers to provide a more networked secure system.

6. Secure Sockets Layer (SSL) and Secure HTTP (S-HTTP)

A Secure Sockets Layer (SSL) is an encrypted protocol developed by Netscape for transmitting private documents over the Internet. Many Web sites use this protocol to obtain a user's confidential information, for example credit card numbers. SSL is designed to prevent eavesdropping, tampering and message forgery.

On the other hand, Secure HTTP (S-HTTP), a modified version of the standard HTTP protocol, is another protocol for transmitting data securely over the Web. The main distinction between SSL and S-HHTP is the former creates a secure connection between a client and a server to ensure any amount of data can be sent securely, whereas the latter ensures individual messages are transmitted securely.

CHAPTER 3 : SIMULATION AND THE EMERGENCE OF WEB-BASED SIMULATION

Simulation modeling, which is viewed as an "indispensable problem solving methodology for the engineers, designers and managers" (Shannon, 1992), has become one of the most powerful tools for the analysis of a wide variety of problems. It has also become an important tool to design new complex processes or systems or to "fine-tube" the performance of existing systems. Law and Kelton in their book, *Simulation Modeling and Analysis,* state, simulation has been found to be a useful and powerful tool for a number of applications, which include designing and analysing manufacturing system, evaluating designs for service organisations, determining ordering policies, designing and operating transportation facilities and analysing financial or economic system (1992). Many surveys have shown as a technique, simulation is one of the most widely used in operations research and management science. Besides, simulation is always ranked second behind "statistical analysis" among other methodologies in terms of its usage and value. Generally, simulation is used as a methodology to describe and analyse the behaviour of a system, where the 'what if' questions are the keys to aid in designing the real system (Banks, 1999).

Historically, over the past forty years of practice, the simulation model created has evolved in many phases. It started with coding in general-purpose languages, then model development in special-purpose simulation languages, model design using higher-level simulation model specification languages and formalisms and comprehensive theories of simulation modeling and environment support for modeling task (Page et al, 1999). Today, technology advancements have enabled a new approach of simulation modeling, and one of them is the concept known as web-based simulation.

3.1 Simulation Overview

Many literatures have given a broad definition of simulation. In general, simulation is defined as a scientific method of constructing and running a model that maps a real system in order to study and investigate its behavior without disrupting the system. A simple explanation of simulation is given by Banks, in which he defines simulation as the imitation of a real-world process or system (1999). On the other hand, a more precise definition of simulation as explained by Shannon is ; "simulation is the process of designing a model of a real system and conducting experiments with this model for the purpose of either understanding the behavior of the system and/or evaluating various strategies for the operation of the system" (1992). In addition, Thesen and Travis describe a simulation as developing a model

to mimic the behavior of a system, thus gain insight into the performance of the system under a variety of circumstances.[1]

3.1.1 Purposes of Simulation

Simulation modeling is an experimental and applied methodology, which the primary purposes are as the following (Shannon, 1992):

- to describe and predict system behavior in a system, which are the effects that will be produced by changes in the system or in its method of application
- to increase understanding of some mechanisms in the studied processes
- to optimize system behavior
- to achieve safe and inexpensive operator training
- to construct theories or hypothesis that account for the observed behavior

On the other hand, McHaney argues that computer simulation can only be used under certain circumstances, which are:

1. The real system does not exist, or the prototype of the system is too costly, time-consuming, hazardous or simply impossible to be built. Some of the examples include airplane, an economic system or a nuclear reactor.
2. The real system exists, but experimentation and testing are expensive, hazardous or disruptive. Some of the examples include a material handling system, a military unit or a transportation system.

---

[1] Thesen, Anne and Laurel E. Travis. "Introduction to Simulation." Proceedings of the 1990 Winter Simulation Conference.

3. A forecasting model is required to analyze an event in long periods of time but in a compressed format. Examples include population growth, forest fire spread or urbanization studies.
4. The real system exists, but experimentation and testing are expensive, hazardous or disruptive. Some of the examples include a material handling system, a military unit or a transportation system.
5. A forecasting model is required to analyze an event in long periods of time but in a compressed format. Examples include population growth, forest fire spread or urbanization studies.
6. Mathematical modeling of the systems has no practical analytical or numeric solution. This might occur in stochastic problems or in non-linear differential equations.

3.2 Steps in a Simulation Study

According to Banks (1999), there are several concepts underlying simulation. These concepts include system and model, events, system state variables, entities and attributes, list processing, activities and delays, and definition of discrete-event simulation. In addition, there are also general steps that guide a model builder in a simulation study (Law and Kelton, 1992 ; Banks, 1999). The following is the summary of the steps that usually involved :

1. Problem definition : each simulation study always begins with a statement of problem

2. Setting of objectives and overall project plan : the objectives indicate issues to be addressed by the simulation study while the project plan include various scenarios to be investigated.
3. Model conceptualisation : the real-world system is abstracted by the conceptual model, which is a series of mathematical and logical relationship concerning the components and structure of the system.
4. Data collection : data collection process can be done simultaneously as the model is being constructed.
5. Model translation : in this stage, the constructed conceptual model is coded into an operational computer form.
6. Verification : in this stage, the operational model is being verified to ensure its proper performance. Simulation analysts are highly encouraged to use and interactive run controller or debugger to aid the verification process.
7. Validation : this is the process where the conceptual model is determined as an accurate representation of the real system.
8. Experimental design : for each scenario to be simulated, decisions need to be made concerning the length of the simulation run, number of replications and manner of initialization.
9. Production runs and analysis : these are used to estimate measures of performance for the scenarios that are being simulated.
10. More runs : determination will be made if additional runs are needed and additional scenarios need to be simulated.

11. Documentation and reporting : documentation is necessary for numerous reasons and the results of the analysis should be reported clearly and concisely.
12. Implementation : the final step of the process is the implementation of the model.

The above conventional steps of simulation will be analyzed in the later part of this chapter to discuss how these typical steps can be supported by the World Wide Web technologies. The following figure depicts the steps in a simulation study as highlighted by Banks (1999).

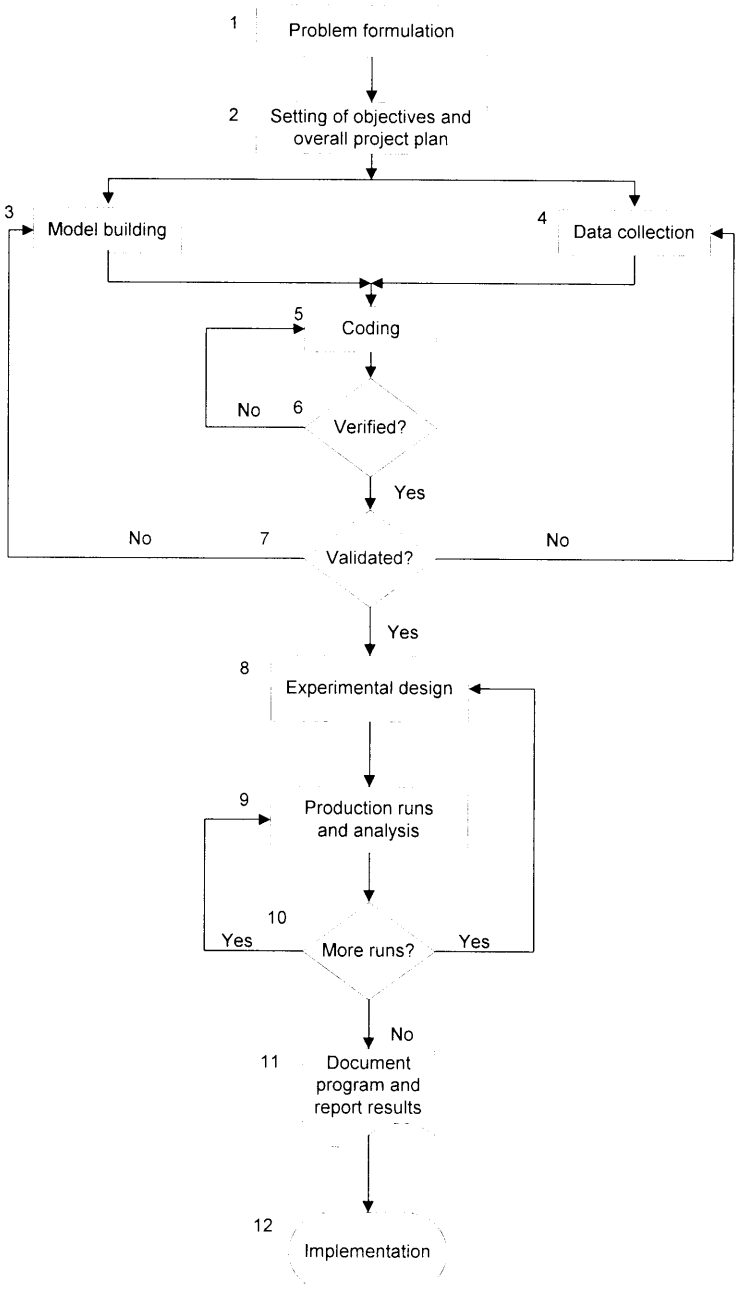

Figure 3- 1: Steps in a Simulation Study

3.3 Areas and Applications of Simulation

Simulation is numerously and diversely applied in many areas. According to Law and Kelton, as a technique, simulation has always turned up either first or second in operations research and management science (1992). Among the representative areas of simulation applications are (Shannon, 1992 ; and Law and Kelton, 1992):

- Computer system : evaluation of hardware components, software systems, networks of hardware, database structure and management, information processing, hardware and software reliability
- Manufacturing : design and analysis of material handling systems, assembly lines, automated production facilities, automated storage facilities, inventory control systems, reliability and maintenance studies and plan layout
- Business : analysis and planning of stock and commodities, pricing policy, marketing strategies, acquisition studies, cash flow and transportation alternatives
- Government : evaluation of military weapons usage, military tactics, population forecasting, land use, health care delivery, criminal justice, and traffic control
- Ecology and Environment : evaluation and analysis of water pollution and purification, waste control, air and pest control, weather prediction, mineral exploration and extraction and solar energy system
- Social and Behavioral : analysis of food and population, educational policies, organizational structure, welfare system and university administration

The following table shows a few areas in which simulation has been applied and possible performance measures are given for each application as highlighted by Thesen and Travis (1992).

Application Area	Performance Measure
Air traffic control	Delays in landing
Bank teller scheduling	Customer waiting times
Cash management	Interest earned
Harbor management	Delays
Location of fire stations	Response times
Parking facility planning	Cars turned away
Computer networks	Delays

Table 3- 1: Some Typical Applications of Simulation

3.4 Web-Based Simulation Overview

Balci and Nance have characterized the evolution of simulation support as reflected both by the "needs push" and "technology pull" (Page, 1996). The necessity of life becomes the catalyst for the first phenomena, or as the adage says 'necessity is the mother of invention'. On the other hand, the latter is the case where invention can be the mother of necessity. Vast innovations of technology advances, such as the creation of the Internet and World Wide Web (WWW), have significantly influenced many aspects of human lives and conventional problem solving techniques. Simulation technique is not an exception to this case. As cited by Page, "web technology has the potential to significantly alter the ways in which simulation models are developed (collaboratively, by composition), documented (dynamically, using multimedia), analyzed (open, widespread investigation) and executed (using

"massive" distribution)" (1996). The deployment of the web technology has resulted in a new method of simulation, a concept known as web-based simulation.

3.4.1 Comparisons of Web-based Simulation to Other Architectures

Comparing web-based simulation to other simulation architectures, the former has more capabilities in supporting various characteristics. Table 3-2 shows characteristics of web-based simulation in contrast to other simulation architectures (Ook, 1999).

Attributes/Characteristics of Effective Simulations	Conventional Package (SIMAN, SLAM, etc)	Distributed Simulation	Object-Oriented Simulation	Web-based Simulation
Ability to allow complex system modeling	√	√	√	√
Ability to facilitate decision support systems	√	√	√	√
Scalability for multiple users		√		√
Ability to facilitate distance-group decision making		√		√
Ability to facilitate online-training and intuitive to use		√		√
Reusability to design for simulation models		√		√
Maintainability to operate with low cost				√
Ability to facilitate GUI design		√		√
Adaptability to real world situations				√
Ability to use multimedia data resources				√
Platform-free simulation modeling				√

Table 3- 2: Characteristics Comparison of Simulation Architectures

In addition, according to Buss, Web-based simulation models must accommodate these characteristics (Page et al, 1999):

1. applications that receive data across the network from a database that will be dynamically determined
2. applications that receive new classes and data unforeseen at the time the model is started
3. applications which use components loosely bound, rather than tightly coupled.

Current Web technologies for instance Java and CORBA, enable traditional simulation to be designed in a more elegant monolith, hence creates a new paradigm for more enthusiastic application.

3.5 Supported Steps in a WBSE

According to Seibt et al, a Web-based simulation environment (WBSE) has to support two different types of simulation applications, which are :

1. the classical steps of a simulation study (as highlighted in previous section)
2. new approaches to modeling and simulation based on cooperative work of distributed web clients and users.

3.5.1 How Web Technologies Can Support Simulation

Accordingly, Web technologies can support a typical simulation study in the following ways :

1. Description of the real system to be modeled and simulated, and definition of the project goals. Web technologies can support these steps by presenting the real objects, processes or services in interactive and dynamic WWW pages.
2. Acquisition of input data by measuring, estimating and calculating values of the real system and determining appropriate distribution functions and their proper parameters. Files transfer with input data can be accomplished easily by the Internet.
3. Creation of logical or artificial model of the real system. The creation of the model can be supported by presenting the relevant information in a HTML page, and allowing the creation of the model to occur on the same page.
4. Verification and validation of the operational model. This step can be executed by the CGI script technique.
5. Planning and running of the simulation experiments with the model. This can be completely done by the CGI technique or Java-based technology like servlet.
6. Processing, compression, interpretation, presentation, recording and comparison of the generated results of the model. The presentation and animation of simulation results in the web can be implemented again by using the CGI technique.
7. Comparison of the simulation results with the real system. The presentation of the results and reports can be done in the Web, or data can be easily transmitted via the Internet.

3.6 WBSE Concept and Components

To determine the concept of a WBSE, Seibt et al emphasize it is important to classify distinct developers and users based on these groups :

- WBSE system designers,
- WBSE components constructors,
- Model designers, and
- Model users.

The architecture of the WBSE is generally compliant to client/server architecture (Seibt et al, ; Iazeolla and D'Ambrogio). Seibt et al have outlined the components of WBSE according to the above groups. These components include CGI scripts for creation of parameter input forms; executable applets for simulation, animation and visualization; database for results recording; and various components to support modeling activities, developing model-user interface, presentations, database access, analyzing input and output data and supporting simulation expert. On the other hand, Iazello and D'Ambrogio illustrate the architecture of the WBSE which compose of a Web simulation server, the client application implemented as Java applet, a set of simulation tool servers, a set of simulation model servers and a CORBA-based infrastructure for interfacing the client and servers. In the study, they concentrate on the uniform and transparent access to simulation models and tools using Java programming language and the CORBA standard. In summary, from the study, the components involved in a web-based simulation architecture are web client, web server, simulation server, simulation tools and other technologies like CORBA, Java and CGI-script. In addition, one of the key features of a Web-based simulation

environment is the deployment of distributed objects and the advent technologies have made the creation and distribution of the objects available, such as creating Applets for the application.

3.7 Approaches To Integrating WBSE and Databases

Looking at the above examples, the studies do not clearly emphasize on the WBSE and databases connectivity. One of the advantages that can be taken from the advent technologies is the integration of WBSE with databases. This architecture offers many potential benefits particularly to the simulation users. The most prominent advantage is simulation models using different input parameters can be easily compared as the results will be stored in the databases and can be retrieved anytime. Hence the 'what-if' analyses can be studied using the stored output. In addition, according to Connolly and Begg (1999) and McFadden et al (1999), database management system (DBMS) offers unlimited potential advantages. In the WBSE context, these advantages may include more information on the same amount of data, improved security, economy of scale, improved data accessibility and responsiveness, increased productivity, improved maintenance through data independence, increased concurrency, simplicity, scalable deployment

In connecting WBSE with databases, several approaches can be taken. One of them is by following the traditional client/server architecture or the two-tiered architecture. However, this environment has some drawbacks, particularly on the issues of scalability, performance and flexibility that make it not suitable for existing

simulation and animation tools. On the other hand, the three-tiered architecture, which can be deployed for the database integration offers several solutions to the problems of two-tiered architecture. However, the complexity of the environment may prohibit the efficiency of the simulation process. As such, the four-tiered architecture, which is the main focus of the study, may be introduced as the resolution to the above approaches. Similar to three-tiered architecture, the four-tiered architecture has resembling components, but there are some additional tools and web technologies to simplify the environment and make it more scalable.

3.8 WBSE and Security Concern

On-line information, publications or applications require good security mechanisms as these mediums are constantly visited by unknown visitors, thus increasing the chances of the servers being penetrated by pedestrian or professional hackers. This is not an exception to web-based simulation as the application is exposed to unlimited users. Unauthorized persons may try to change configuration, execute current files illegally and modify existing data. To ensure a secure information transaction in a web-based simulation environment, it should have these characteristics (db2.travel.ch):

- Confidentiality : the contents of the data remains private as they pass through the Internet. Encryption ensures confidentiality.
- Integrity : the contents are not being altered while being transmitted. Without integrity there is no guarantee that the data sent matches the data receive. Encryption and digital signature ensure integrity.

- Accountability : both the sender and receiver agree that the exchange took place. Without accountability, the addressee can claim the message never arrived. Digital signature also ensures accountability.
- Authenticity : based on trustworthiness and without authenticity, there is no way to be sure that anyone is who they say they are. Authentication ensures authenticity.

To build and maintain a secure web-based simulation architecture, there are several elements within the network topology that have to be closely examined. These are routers, firewalls and servers, which include file, database, applications Web, mail and others (Kessler, 2000:a). Each of the elements within the Web architecture must have a "secure synergy" with the others as there will always be a weak point that attackers will find and exploit. To logically know where to locate services, the network can be partitioned through a demilitarized zone or DMZ. In addition, the first set of defenses should be implemented to the router as it is the first point of presence to the Internet (Kessler, 2000:b) using the packet filter and other access control capabilities.

The primary system that protects internal, trusted, private network systems from the outside, less trusted world is known as firewalls. The firewall can supplement the access controls implemented in the router and has more intuitive user interface (Kessler, 2000:b). The most common type of firewall that can be applied for a web based simulation environment involves basic packet filtering and access control with more intelligence capabilities than the typical router.

In addition, Kessler adds, trust is the main concern in this environment, and it must be established between communicating parties (2000:b). In particular, users or clients need to be sure that :

- They are communicating with the right server
- What they send, ie the data is delivered unmodified
- They can prove that they sent the data
- Delivery is guaranteed

Meanwhile, the servers need to be sure that :

- They are communicating with the right client
- The content of the received message is correct
- The identity of the user is unmistakable
- They acknowledge receipt of the message.

CHAPTER 4 : ESTABLISHING A CONCEPTUAL FRAMEWORK FOR A FOUR-TIER WEB-BASED SIMULATION ENVIRONMENT

As it has been highlighted in Chapter 1, the study undertaken introduces a framework for a four-tier Web-based simulation environment. The main purpose of the framework is to serve as a general guideline for future development of the architecture. In designing the architecture, Rapid Application Development (RAD) methodology is applied, in addition to referring to other scholars' work towards Web-based simulation applications.

4.1 Rapid Application Development

When a system is developed, it is important to use a methodology to design it (Schach, 1997). Applying a set of methods in designing a system will ensure a consistent, reproducible approach is applied, reduce the risk associated with shortcuts and mistakes and result in improved performance and user satisfaction (superstats.com). There are various types of methodologies that can be practiced when a system or project is implemented, and for this study, Rapid Application Development (RAD) methodology is used.

RAD (Rapid Application Development) refers to a development lifecycle designed to give much faster development and higher quality results than the traditional lifecycle. In other words, it is a methodology for compressing the analysis, design, build and test phases into a series of short, iterative development lifecycles. The diagram below illu strates the application of RAD as compared to traditional method in developing a system lifecycle.

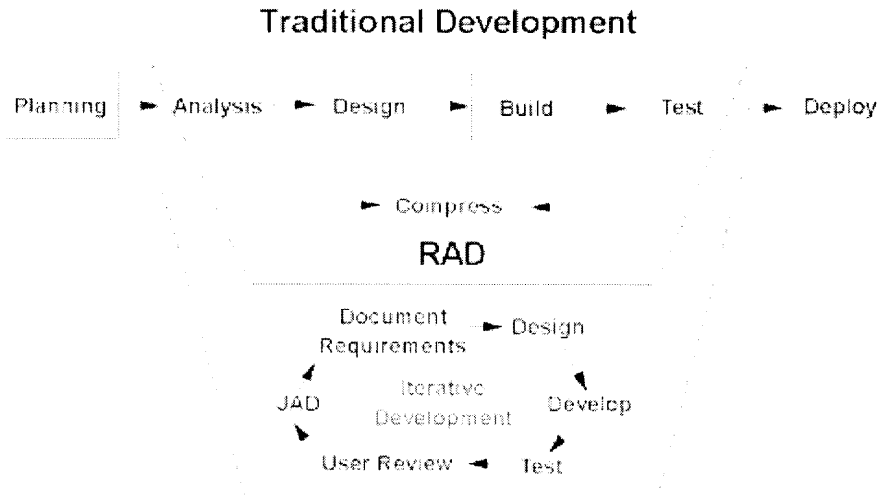

Figure 4- 1: Traditional Development vs. RAD

Many projects have demonstrated the application of RAD in their system development is superior to the traditional development. This is due to the iteration of development lifecycle, which allows for effective and self-correction. A fundamental principle of iterative development is that, each iteration delivers a functional version of the final system. It is a properly engineered, fully working portion of the final system and is not the same as a prototype.

The RAD lifecycle has four phases as illustrated in the following diagram.

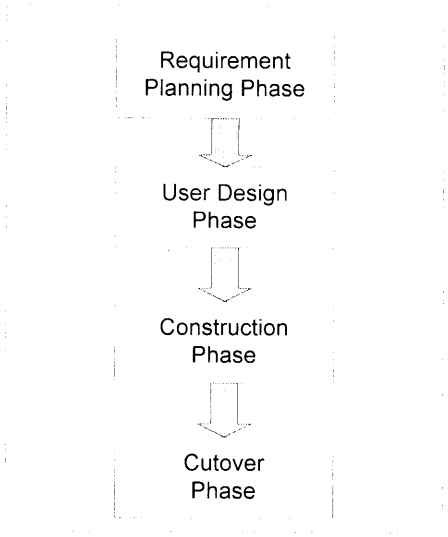

Figure 4- 2 :RAD Four Phases

Each phase or iteration in the RAD methodology adds value to the solution and incorporates lessons learned from the preceding iteration. However, due to some constraints and limitations, the current study will only be implemented up to the design phase.

4.2 Approaches to Designing the Framework

In designing a Web-based simulation, many approaches can be taken based on the focus of the study. For instance, Guru, Savory and Williams (2000) look at the Web-based interface or toolkit that can be developed for storing and executing SIMAN simulation models over the Internet. The toolkit consists of a WWW interface to the simulation tool and a Web-accessible database for model storing purpose. The study performed by them is based on a three-tier model architecture, which consists of an application server, a database server and a SIMAN server, and all the servers are

linked by the toolkit. However, the study does not give a detail information on the types of the Web tools used, and it just focuses on the overall process diagram.

Another related study has been conducted by Syrjakow, Berdux and Szczerbicka (2000). The main focus of the research is to develop interactive multimedia elements for Web-based learning materials, and animations are added to the application. The animations are built as Java-applets and the study takes the advantages of the technologies offered by Java due to their numerous benefits. Yet, the research only concentrates on the front-end application to the users.

On the other hand, Iazeolla and D'Ambrogio put the effort in developing a Web-based simulation environment based on Java and CORBA-based architecture. They develop a prototype environment for queuing network simulation models implemented in the language of the QNAP2 simulation tool. Again, the study uses Java and CORBA-based architecture due to the benefits offered by them. Kuljis and Paul (2000) although do not develop any model or application, they give a review on various new technologies that can be applied for discrete-event simulation software development. These include CORBA and Java component technology JavaBeans. According to them, possible benefits of using JavaBean include platform neutrality, Internet capability and visual programming.

In addition, Cholkar and Koopman (1999) in their research on deployable Web-based network simulation framework focus on a large-scale Web-based simulation

deployment, which uses a set of standard CORBA-IDL based programming interfaces. A prototype application developed y them demonstrates that the CORBA components not only provide language and platform-independence, but also provide the ability for simulationists to connect objects to a third party distributed simulation. The uniform messaging approach enables objects to be reassigned to different simulation entities without requiring code modifications, hence it attains extensibility and flexibility. In their study, they develop the architecture of the network simulation framework using Object Oriented (OO) view of a network. By using this approach, the core components of the communication networks are identified, and based on these core components, the software counterparts are built. The hardware components are derived from an abstract base class called *Entity*, and each entity consists of a *module manager*, which encapsulates an API, known as the 'Module Plug-in API' (MPA). The MPA and Module Callback Interface (MCI) are flexible enough as they provide standard interfaces semantics for simulation entities, and the flexibility is further extended by CORBA IDL. From the entity module, they implement a prototype framework using a traditional two-tier client-server model. The prototype consists of a simulation kernel implemented in C++ and a user interface written in Java. To provide remote object invocations and facilitates communication between the user-interface and the simulation kernel, a CORBA ORB is used. In addition, a Relational Database Management System (RDMS) is used to provide the necessary logging facilities. The system architecture of the implementation is shown in the following figure.

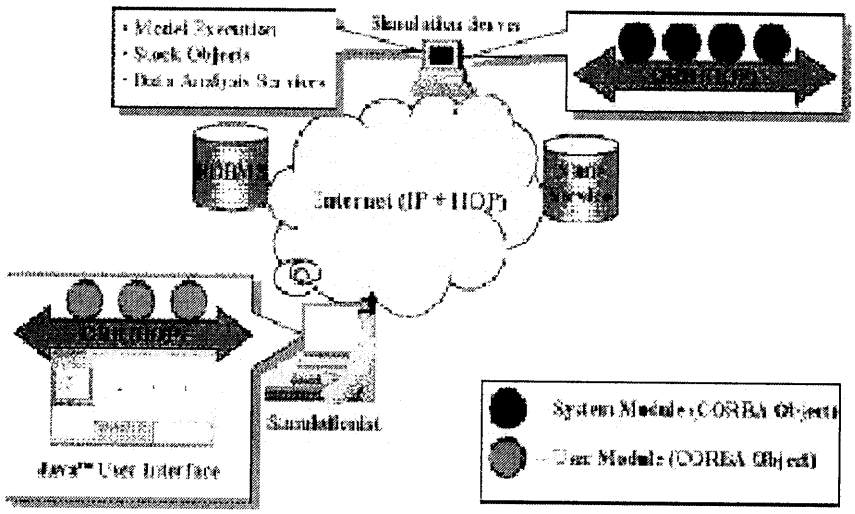

Figure 4- 3 : System Architecture of the Prototype

Typical sequences of interactions that take place among the various components in the system are shown in the next diagram. The work presented by Cholkar and Koopman demonstrates that it is possible to attain a Web-based network simulation framework suitable for large-scale deployment using the CORBA-IDL based programming interfaces without sacrificing extensibility and flexibility.

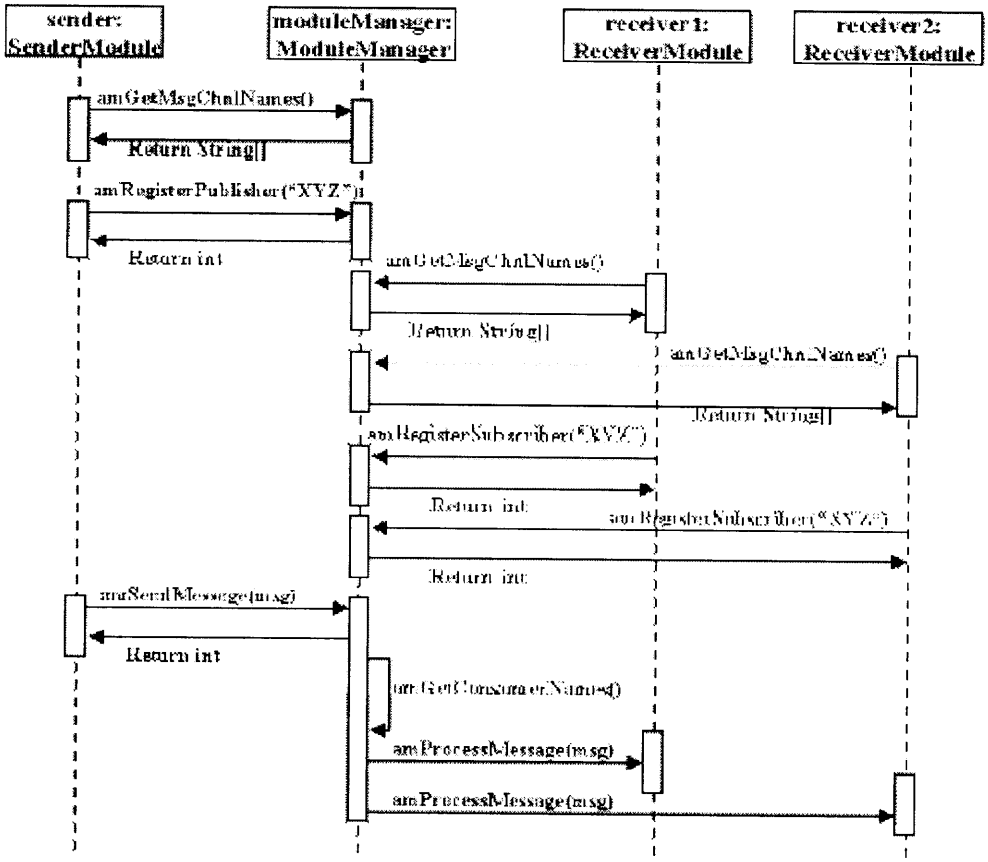

Figure 4- 4 : Sequence Diagram for a Typical Startup Scenario

Hence, based on previous research done on Web-based simulation and available current technologies, the study to design a conceptual framework for a four-tier Web-based simulation is developed. The study takes a full advantage of the Java technology based on the potential benefits. Among others include (Luo et al, 2000):

- Java has built-in support for producing sophisticated animations
- Java has built-in threads making it easier to implement the process/resource interaction worldview
- Models implemented as Java applets can be made widely accessible through Web browsers

- Java's universal portability eliminates the need to port to a different platform, to recompile or to relink.

In addition, the framework is built based on ideas from CORBA to create an independent, distributed network simulation as it allows standard communication between objects in a distributed environment.

4.3 Designing a Conceptual Framework for a 4-Tier Web-based Simulation Environment

The main objective of the study is to design a conceptual framework for a 4-tier Web-based simulation environment. Hence, several tools and technologies will be identified to integrate the devices for the system. In this case, most of the tools chosen are Java-based technologies since they are more robust and portable for development, deployment and management of Web applications. However, no specific simulation tool has been identified or designed for the study, and it is assumed the simulator is built from Java technologies for easy development and support. No specific product has also been identified for the study, for instance gateway connector products such as Lotus® Domino™ and Net.Data®, or operating system platforms such as Windows NT™ and Sun Solaris as the main purpose of the study is to identify tools and techniques for Web-based simulation environment integration.

The following table defines the cycles that will be followed in designing the conceptual framework for the WBSE environment.

Loop	Input	Functions Added	Output
0. Generic Design	Existing section	N/A	Baseline design document
Baseline 1		**Element Management**	**Basic Web Equipment**
1. Graphics Design	Network Topology	Graphics Design Implementation	Graphical Model of Proposed WBSE
2. Develop System Interfaces	Output from 1	System rule development, system graphics development	Integration of the system
Baseline 2		**Network Management**	**Web-based Simulation Network Architecture**
3. Integrate Web tools, Internet	Output from 2, plus additional techniques	Integrate Web tools and other equipment	Improved fit with other devices/components
4. Integrate security tools	Output from 3, plus additional techniques	Integrate security tools and other equipment	Improved fit with other devices/components
Baseline 3			**Complete Web-based Simulation Network Architecture**

Table 4- 1 : System Life-cycles

However, as mentioned before, the undertaken study will be carried out up to the design phase only, and not until the construction and cutover phases. In addition, the above sequence is just an initial plan of the work for the purpose of this study. One of the features of RAD methodology is that lessons learned from one phase may have a knock-on effect and the development plan must be flexible enough to react to changes and feedback. Thus, should any changes need to be done prior to the implementation in the future, the plan should be revised to reflect the requirements.

4.3.1 Baseline 1

4.3.1.1 Graphics Design

A graphical model of the proposed Web-based simulation environment is depicted in the following diagram. Since the application of the simulation is based on the Web-environment, the system will use Java applet which acts as the distributed object that implements the client application via a CORBA-based infrastructure.

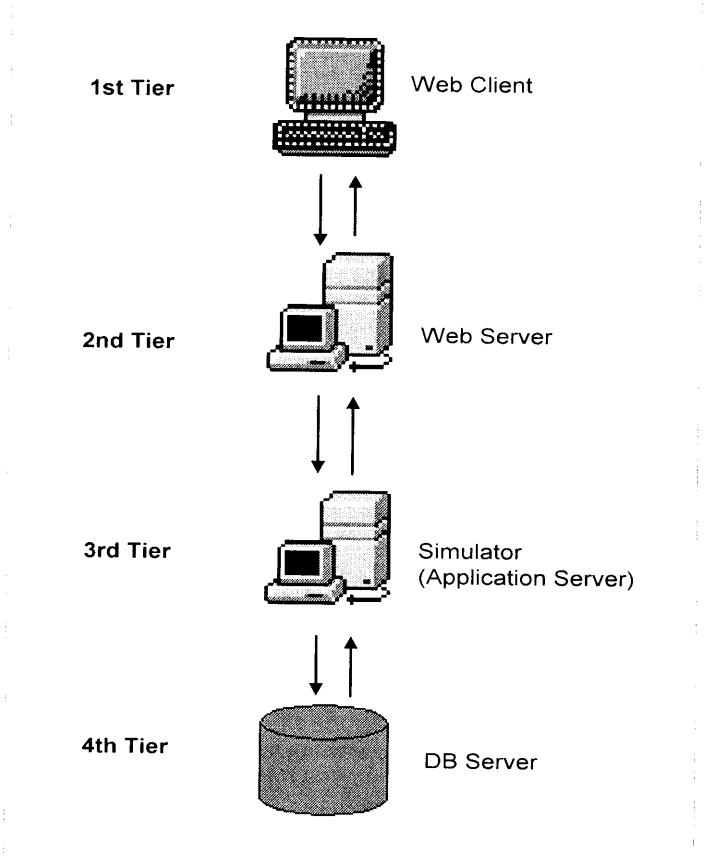

Figure 4- 5 : A General Layout of the 4-Tier Web-based Simulation Environment

The basic components of the system architecture consist of web clients as the first-tier, which act as the front-end system or system interface to the users, web server as the second-tier, the application server which the simulation tool resides as the third-tier and finally, the database server which records data and simulation results as the fourth-tier. Details of the system integration will be discussed further in the next section.

4.3.1.2 System Interfaces

To integrate the components as one whole system, several tools and techniques need to be incorporated. These include tools to integrate the Web client with the Web server, the Web server with the application server and the application server with the database server. In addition, since the environment involves applets which implement client application, the plug and use capabilities should also be provided for the system. To simplify the integration, the following diagram models the enhanced environment.

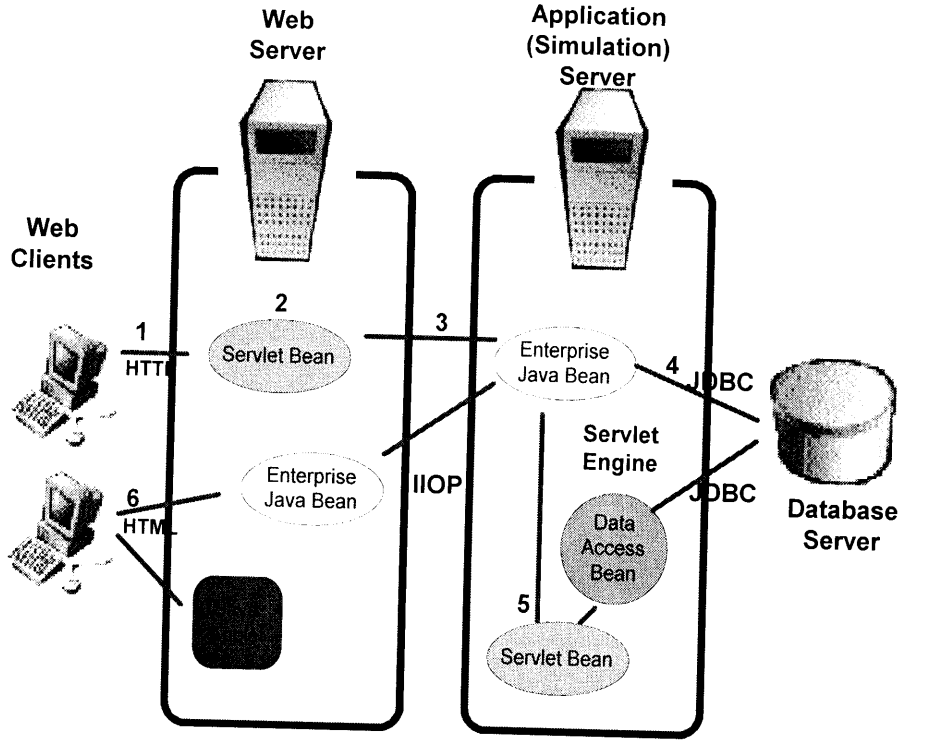

Figure 4- 6: Enhanced Model of the Proposed WBSE

There are several general basic steps involve in the process:

1. The client sends a request to the Web server, which could be to obtain the simulation application, create a logical model of the real system, verify and validate the operational model or query for simulation information, such as obtaining previous simulation results.
2. The Web server identifies the requests by addressing a servlet and utilizes its configuration data to pass the request on to the runtime engine code. The appropriate servlet is invoked via HTTP.
3. The servlet conducts the query while connecting to the application server and the transaction is performed.

4. Results will be saved and sent to the database server.
5. When the results are sent back to the server, the servlet regains control and manages the response generation.
6. The computed Web page with the results of the query is then served back to the Web client via the Web server using HTML.

1. The Client Application

The client application, which is implemented as Java applets, is run by any Java enabled Web browser such as Netscape Navigator and Microsoft Internet Explorer using standard HTTP and HTTPS (SSL) protocols. The main task of the client application is to provide a uniform and friendly-user interface for users to obtain services provided via the underlying CORBA infrastructure. Other tasks include retrieving the simulation tool, input parameters or data, run simulation models, save the simulation models and obtain and view the simulation results.

2. The Web Server

At its core, the Web server is a file server. It responds requests from the Web client and sends the requested files back to the client. The Web server communicates with the application server via servlet engine. In the proposed environment, the Web server is placed outside the firewall, yet workload management can still be implemented across Enterprise Java Bean (EJB) inside the firewall. The firewall helps filter IIOP as well as TCP/IP for security reasons. In addition, servlet is also

used to access the application server via command object, which provide an interface to different commands in the application.

3. The Application/Simulation Server

In this environment, the application (simulation) server provides a middle layer between the resource and the Web server. The simulation software is encapsulated into the environment with a technique called wrapping. CORBA provides a standard object-oriented approach for encapsulating the software (Pope, 1998). The wrapper is reusable to encapsulate the simulation models when it is implemented according to the CORBA standard, hence providing the environment a plug and use capability. In addition, the application server must be capable of supporting the accessed system. This can be done via gateway connectivity that converts the HTML requests into sets of parameters. Although gateway connectivity is not scalable in nature, it still provides a simple to-use and easy-to-develop alternative, which can be called from servlets.

4. The Database Server

In the Web environment, the database servers works similarly to the way the server works in a more traditional client/server architecture. The server responds to requests and return information via Call Level Interface (CLI), which is JDBC. JDBC drivers are used for database access from servlets. In addition, the database server must have an open architecture approach to allow interoperability with variety of systems and

technologies. Besides, it should employ acceptable performance and require minimal administrative overhead.

The application or the simulation server can communicate with the Web server and the database server through distributed objects. In this case, a CORBA-based infrastructure, which is the IIOP (Internet Inter-ORB Protocol) Bus and Wrappers are used to interface the application over the Internet. The IIOP specifies how GIOP (General Inter-ORB Protocol) messages are exchanged over a TCP/IP network, thus making it possible to use the Internet as a backbone through which various ORBs can communicate (Pope, 1998). The following figure illustrates the specification of the wrapper to access the services.

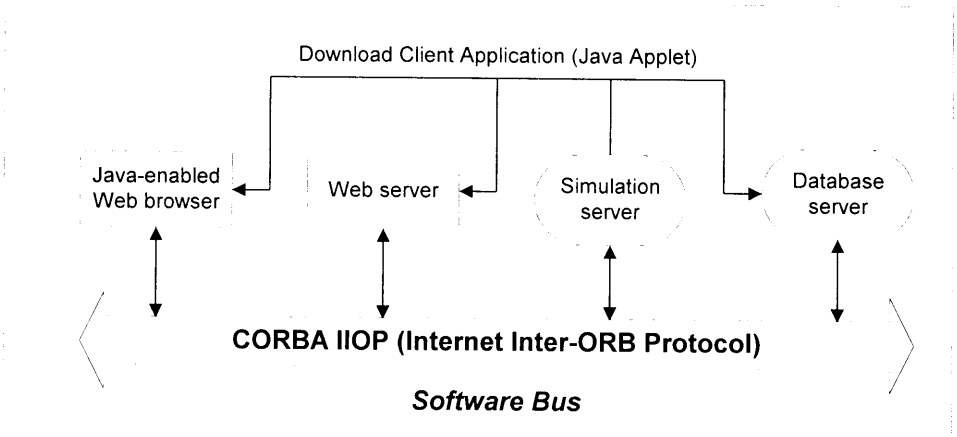

Figure 4- 7 : CORBA Infrastructure in the WBSE

4.3.2 Baseline 2

4.3.2.1 Network and Web Tools Integration

A detailed architecture of the Web-based simulation environment is illustrated in the following diagram.

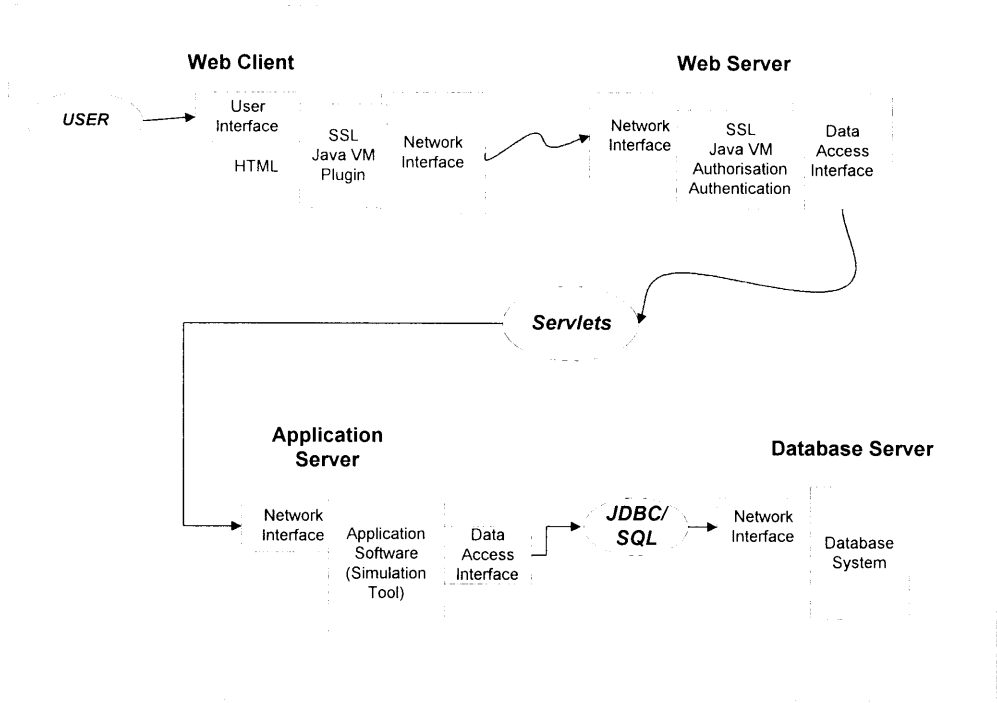

Figure 4- 8: A Detailed Architecture of the WBSE

In addition, looking form the networking perspective, there are relationship attributes that allow objects behavior to be defined in relation to other objects. The class definition of the objects includes the following relationship.

Relationship	Converse	Description
Connects	ConnectedBy	Relationship between circuits and their component devices. A circuit 'Connects' several pieces of equipment; equipment is 'ConnectedBy'a circuit.
Contains	ContainedIn	Describes the physical containment of a piece of equipment. Used to 'bubble up' alerts to a higher level.
MountedOn	Mounts	Relationship between I/O Ports and the card they are mounted on.
Networks	NetworkedBy	Relationship between a sub-system and the equipment that form the sub-system.
OwnedBy	Owns	Equipment such as servers are 'OwnedBy' a system.
Pings	PingedBy	Relationship between 'pinger' and the 'pinged devices' that it pings.
PipedBy	Pipes	Relationship between a channel and the path in which it is contained.
UsedBy	Uses	Relationship between a sub-system and the services that use the sub-system.

Table 4- 2 : Object Relationship

The above relationships allow object behavior to be correlated with other items. For instance, if a device reports a fault, the device can be can be related to the circuit it is 'ConnectedBy'. This circuit can then be related to all the devices in 'Connects' to determine if the root of the problem lies elsewhere in the network. The following figure shows the relationships among the classes using the above relationships.

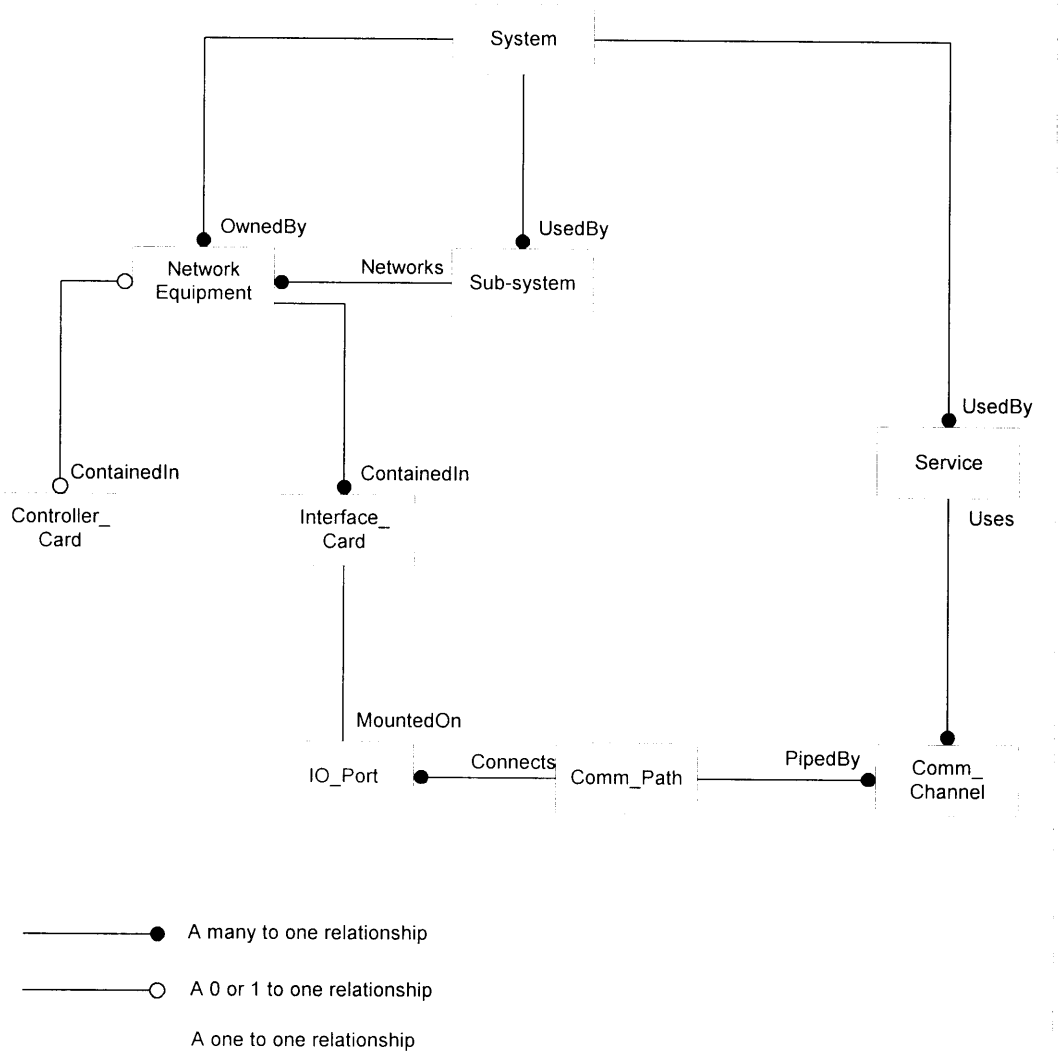

Figure 4- 9 : Relationships Among Classes

The above relationships are designed to make the integration of devices, for instance to connect a server to another easier. In designing an architecture, the networking element is important to ensure the configuration of the system is done properly.

4.4 Towards the Implementation

Although the main focus of the study is to design a conceptual framework for a four-tier Web-based simulation environment, it is more appropriate if the general overview of how the implementation will be performed is given. By giving the general outline of the implementation, future development can be easily conducted, hence reduce the complexity of the work.

The most fundamental element of the architecture is simulation models implemented as distributed objects. As discussed before, the study does not concentrate on building a Web-based simulation tool, which is similar to Silk or SimJava. However, it is assumed the simulation model is designed based on Java programming language. The application is then implemented as Java applets so that it can be posted to the Java-enabled Web browser to be executed by users. In addition, when the user wants to compile, execute or debug a model, an HTML document will be created, and this document will contain the model frame, and model ID as form elements.

The transparent communication among the objects and the Web browser is performed by CORBA IIOP Software Bus, as highlighted in the previous section. CORBA provides a flexible, extensible, platform-independent and language independent environment that is suitable for large-scale deployment (Cholkar and Koopman, 1999). In addition, to integrate the application server to the database server, Java Database Connectivity is used. It acts as the driver to ensure the

application is able to communicate with the data source. Details of JDBC in the context of the study is outlined below:

4.4.1 Java Database Connectivity

One of the key distinction of the proposed Web-based simulation environment is its integration with database. In addition, it is vital to ensure the application is able to communicate with the data source. Hence, an open, portable, platform-independent database solution, which is Java Database Connectivity (JDBC) has been chosen to access the database. In this proposed environment, JDBC is used to do three main tasks as illustrated in the diagram below:

1. establish a connection with the database
2. send SQL statement
3. process the results

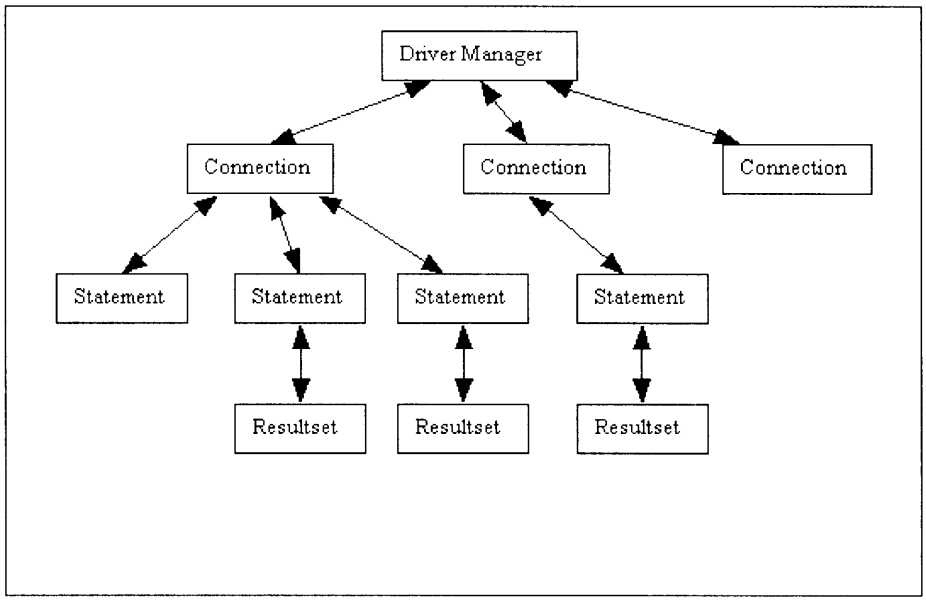

Figure 4- 10 : The JDBC API's Tasks

- Database Connectivity Object Classes and Relationship.

The database application for the environment has been broken down into smaller components to enable independent design, without affecting the rest of the system. The components identified for the database are shown in the following diagram.

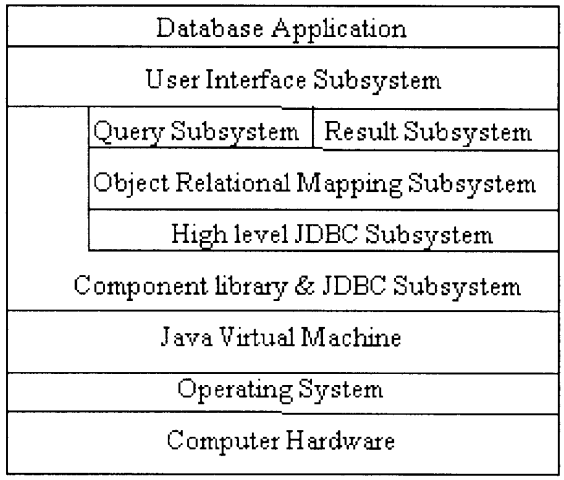

Figure 4- 11 : Block Diagram of Database Application

In terms of Java classes, the JDBC API consists of:

- java.sql.Environment - allows the creation of new database connections;
- java.sql.Connection - connection-specific data structures;
- java.sql.Statement - container class for embedded SQL statements;
- java.sql.ResultSet - access control to results of a statement.

The database connectivity objects can be classified into these classes:

Database Connection.	Attribute String.
Database Resultset.	Attribute Int.
SQL Statement.	Attribute Double.
Mapping.	

In addition, the relationships derived from the classes are:

SQL Statement Uses Database Connection.

Database Resultset describes SQL Statement.

Mapping object generates SQL Statement.

Mapping object contains String, Int and Double.

The tasks that the DatabaseConnect class carries are as the following:

1. open a database connection
2. enable SQL command to be executed
3. enable results to be retrieved and viewed based on search criteria.

The DatabaseConnect class will operate on top of the JDBC API and it uses classes Connection, Statement, ResultSet and ResultSetMetadata. The next process of the connectivity will involve the DatabaseConnect implementation to enable a connection to be opened, queries to be executed and results to be returned. The following shows the data types supported in JDBC

```
CHAR String Single Character
VARCHAR String Variable length string of characters
LONGVARCHAR java.io.InputStream Very long (multi-megabyte)
strings
NUMERIC java.sql.Numeric Absolute precision fixed-point values
DECIMAL java.sql.Numeric Absolute precision Decimal value
BIT boolean Single bit/binary value (on or off)
TINYINT byte 8-bit integer
SMALLINT short 16-bit integer
INTEGER int signed 32-bit integer
BIGINT long signed 64-bit integer
REAL float Floating-point value
FLOAT float Floating-point value
DOUBLE double Large floating-point value
BINARY byte[] Array of binary values
VARBINARY byte[] Variable length array of binary values
```

```
LONGVARBINARY java.io.InputStream Very large (multi-megabyte)
array of binary values
DATE java.sql.Date Date value
TIME java.sql.Time Time value (GMT)
TIMESTAMP java.sql.Timestamp Time value with additional
nanosecond field
```

4.5 Security Tools Integration

One of the most prominent issues in any Web-based application is the security. It is highly concerned with protecting the internal network from the outside network. From the Web-based simulation environment perspective, security is concerned with protecting the managed object. It provides the rules for authentication procedures, and the maintenance of access control routines. Firewall is the proposed security feature for the environment and as a security gate, it provides security to components inside the system gate, as well as control of who (or what) is allowed to get into this protected environment, as well as those allowed to go out. It works like a security guard at a front door, and will be set up to provide controllable filtering of network traffic.

There are several types of firewall techniques and the one chosen for this environment is packet filter. Packet filtering systems will allow or block certain types of packets, and the type of router that can be used in the packet filtering firewall is screening router. The router knows all the information that contain in the sets of header in each packet passing through the firewall, and the main information contain is :

- IP source address

- IP destination address
- Protocol
- TCP source phone
- TCP destination port

Besides, the router knows about the packet's information that is not reflected in the packet headers such as the interface the packet arrives on and the interface the packet will go out. By deploying the security feature in the environment, the system is able to block all connections to or from certain distrusted systems. By using firewalls, new connections must be authenticated and approved at the application layer (this has to be referred to OSI Model for greater details). For instance, new users should be asked for passwords and once validated, they are permitted to access the system. Once this has been done, the remainder of the connection is passed down to the session layer, where the packet filter watches the connection to ensure that only packets that are part of the ongoing (already authenticated and approved) conversation are being passed.

Although at a glance, a Web-based simulation application seems not to have any critical information or data to be protected from, having an on-line application that is widely exposed to many users, requires a close look at the security. This is because one never knows what the intention of the user is.

CHAPTER 5 : CONCLUSION

Web-based simulation is a new birth of the conventional simulation and its emergence is made possible by the rapid advancement of technologies. However, due to its fresh commencement, a lot of research and work can be conducted to build a more dynamic and boundless Web-based application. In addition, efforts should be more focus on exploiting the Web technologies as the Web particularly has strong impacts to end-users as it can increase group collaboration in decision making.

In designing a Web-based simulation environment, various approaches can be taken. One of them is integrating the architecture with databases. In this study which focuses on the Web-based simulation-database approach, a conceptual four-tier architecture is built since it is more efficient and scalable compared to two or three-tier architecture. Java-based technologies are chosen since they are more portable, robust and reusable. The techniques used in the study include using CORBA IIOP Bus and Wrappers infrastructure as the backbone interface for the distributed objects over the Internet. Besides, Java Database Connectivity is used as a driver to connect the application to the database server. In addition, security is also concerned in

designing the architecture since the Web-based application is more open to external users, whom some of them can be distrusted. In ensuring a more secure communication, a firewall is integrated into the system to protect the internal network from the external network. The firewall acts as a gate to safeguard the components inside the gate system, as well as to control who or what is allowed to enter the protected environment.

However, due to some limitations, the proposed environment is unable to be implemented at this time of study. Yet, efforts are being taken to further design the architecture in more detail, and these include :

1. to specify the type of Web-based simulation application for more detailed integration purposes
2. to design the architecture in more details based on each tier requirements and specifications
3. to further study the Web technologies for enhancement purposes
4. to better design the security countermeasures

REFERENCES

"Applied Business Solution." 2000 <http://www.superstats.com/>

Banks, Jerry. "Introduction to Simulation." Proceedings of the 1999 Winter Simulation Conference, pp 4-13 (1999). P. A. Farrington, H. B. Nembhard, D. T. Sturrock and G. W. Evans (eds)

Bates, Bud. "Hands-on Client/Server Internetworking". McGraw-Hill. 1998

Beyda, William J. "Data Communications : From Basic to Broadband". Prentice Hall (1996)

Birman, Kenneth P., "Building Secure and Reliable Network Applications". Manning, 1996.

Bottjer, Mark. "Integrating Databases with the World Wide Web" (1997)

Cheng, Josephine and Susan Malaika. "Web Gateway Tools : Connecting IBM® and Lotus Applications® to the Web". John Wiley and Sons, Inc. 1996

Cholkar, Arjun and Philip Koopman. "A Widely Deployable Web-based Network Simulation Framework Using CORBA IDL-based APIs". Proceedings of the 1999 Winter Simulation Conference, pp 1587-1594 (1999). P. A. Farrington, H. B. Nembhard, D. T. Sturrock and G. W. Evans (eds)

"Cisco Networking Academy Program. Cisco System". 2000 (CD)

Cobb, Edward., James Hamilton and Geoff Sharman. "Do I Need a Transaction Processing Monitor and a Database?" 1996 < http://www-4.ibm.com/software/ts/cics/library/whitepapers/dbtmgrs/>

Connolly, Thomas M., and Carolyn E. Begg. "Database Systems : A Practical Approach to Design, Implementation and Management". Addison-Wesley. 1999

"Developing a Database Connectivity Framework for Intranet/Internet Applications" 2000. <http://www.esatclear.ie/~chumor/TitlePage.html>

Dodds, Tom., Warren Kerby and Michael Howard. "Data Security and Data Availability for End Systems" 2000 <http://www.microsoft.com/technet/security/datavail.asp>

El Sheikh, Asim A.R., et al. "A Microcomputer-Based Simulation Study of a Port." Opl.Res. Soc. Vol 38, No 8, pp 673-681 (1987).

Freeman, E. "Link Everything to Anything" 1996 <http://www.datamation.com/PlugIn/issues/1996/oct/10cs1html>

Guru, Ashu., Paul Savory and Robert Williams. "A Web-based Interface for Storing and Executing Simulation Models" Proceedings of the 2000 Winter Simulation Conference, pp 1810-1814 (2000). J.A Joines, R.R. Barton. K.Kang, and P.A. Fishwick (eds)

Hinrichs, Randy. "Intranet 101: A Guide For Intranet Newbies" Intranet Journal (1996) <http://www.intranetjournal.com/newbie.html>

"History of the Internet and the World Wide Web." 2000 <http://wdvl.com/Internet/History/>

Iazeolla, Giuseppe and Andrea D'Ambrogio. "A Web-based Environment for the Reuse of Simulation Models" <http://www.informatik.unibw-muenchen.de/SCS/confernc/wmc98/websim/wbms/d7/iazeollag.htm>

"Integrating Databases With Java Via JDBC" Java World, May 1996.

"JDBC Data Access API". 2000. <http://www.javasoft.com/products/jdbc/index.html>

Kessler, Gary C. "Securing Your Web Site" 2000:a <http://www.garykessler.net/library/web_security.html>

Kessler, Gary C. "Web of Worries" 2000:b <http://www.infosecuritymag.com/apr2000/websecurity.htm>

Koh, Peng-Hong, et al. "Using Simulation to Preview Plans of a Container Port Operations." Proceedings of the 1994 Winter Simulation Conference, pp 1109-1115 (1994).

Kuljis, Jasna,. and Ray J. Paul. "A Review of Web-based Simulation : Whither We Wander?" 2000. Proceedings of the 2000 Winter Simulation Conference, pp 1810-1814 (2000). J.A Joines, R.R. Barton. K.Kang, and P.A. Fishwick (eds)

Law, Averill M,. and Micheal G. McComas. " Secrets of Successful Simulation Studies." Proceeding of the 1991 Winter Simulation Conference (1991).

Law, Averill M. and W. David Kelton. "Simulation Modeling and Analysis." New York : McGraw-Hill, Inc. 1992.

Leiner, Barry M., et al. "A Brief History of the Internet". 2000 <http://www.isoc.org/internet-history/brief.html>

Luo, Yuh-Chuyn et al. "Distributed Web-based Simulation Optimization". Proceedings of the 2000 Winter Simulation Conference, pp 1785-1793 (2000). J.A Joines, R.R. Barton. K.Kang, and P.A. Fishwick (eds)

"Making Your Communication Secure." 1997 <http://db2.travel.ch/Docs/icswg011.html#Header_737>

McFadden, Fred R., Jeffrey A. Hoffer and Mary B. Prescott. " Modern Database Management". Addison Wesley Longman, Inc. 1999.

McHaney, Roger. "Computer Simulation : A Practical Perspective" Academic Press, Inc. 1991.

Orfali, Robert, Dan Harkey and Jeri Edwards. "Client/Server Survival Guide". John Wiley & Sons, Inc. 1999

Öszu, M. Tamer and Patrick Valduriez. "Principles of Distributed Database System". Prentice Hall. 1999.

Page, Ernest H., et al. "Web-based Simulation : Revolution or Evolution?" 1999 <http://ms.ie.org/page/papers/tomacs/webPanelPaper/paper.html>

Pope, Alan. "The CORBA Reference Guide : Understanding the Common Object Request Broker Architecture". Addison Wesley. 1998.

"Resources : A Comprehensive Resource for Publishing on the World Wide Web." 2000 <http://www.webcom.com/webcom/help/overview/www.shtml#INTER>

Schach, S. "Software Engineering With Java". Times Mirror Education Group, Inc. 1997

Seibt, Frank., Marco Schumann and Jurgen Beikirch. "Concepts and Components for a Web-based Simulation Environment". http://isgsim1.cs.uni-magdeburg.de/seibt/sandiego/sandiego.html

Shannon, Robert E,. "Introduction to Simulation." Proceedings of the 1992 Winter Simulation Conference. (1992)

Syrjakow, Micheal., Helena Szczerbicka and Joerg Berdux. "Interactive Web-based Animations for Teaching and Learning" (2000). Proceedings of the 2000 Winter Simulation Conference, pp 1651-1659 (2000). J.A Joines, R.R. Barton. K.Kang, and P.A. Fishwick (eds)

Technical Overview of Netscape Application Server 4.0. 2000. <http://www.iplanet.com/products/whitepaper/whitepaper_3.html>

Teleen, S. (1996) Intranets and Adaptive Innovation. http://www.amdahl.com/doc/products/bsg/intra/adapt.html

"The Most Complete CISCO-Networking CBT Essentials" 2000. Vol 2. Cisco Powered Network. (CD)

Thesen, Arne and Laurel E. Travis. "Introduction to Simulation." Proceedings of the 1990 Winter Simulation Conference. (1990)

Thesen, Arne and Laurel E. Travis. "Simulation for Decision Making" West Publishing Company (1992), p3

Tristram, C. "Middleware Really Makes Client/Server Applications Really Work". 1996 <http://www.datamation.com/PlugIn/issues/1996/augsoft2.html>

Whicker, Marcia Lynn and Lee Sigelman. "Computer Simulation Application : An Introduction." Applied Social Research Methods Series. Vol 25. SAGE Publications. (1991)

Yarborough, William J., "Building Communication Networks with Distributed Objects". McGraw-Hill, 1998.

GLOSSARY

Applets	Programs executed by the Web browser and cannot be outside of the browser's own window.
Application server	A device that work with the Web application, and acts as a middle layer between a Web server and a resource or database server.
CORBA IIOP	A remote method protocol that enables a Web browser to access remote objects located on a server.
Database server	A device that stores data, responds to requests and returns information.
Distributed objects	Objects used to connect heterogeneous databases across different platform.
Enterprise Java Beans	Java components, which are easily reused and have better scalability compared to Transaction Managers.
Internet	The world's largest computer network, which acts as a mechanism for information dissemination and a medium for remote collaboration and interaction.
JDBC	A database access Application Programming Interface (API) that enables developers to write Java applications that access databases.
Network security	The protection of the network against intentional or unintentional threats using computer-based or non-computer-based controls.

Object	A uniquely identifiable entity that contains both attributes to describe the state of a 'real world' object and actions associated with it.
Rapid Application Development	A system development methodology that emphasizes on iteration lifecycles.
Servlets	Java programs used to communicate with databases, do necessary business logic and provide content
Simulation	A method used to describe and analyze a behavior of a real world process or system, where the 'what if' questions are the keys in designing the real system.
Tiered-architecture	A system which the tasks are separated based on different platform and layer.
Web client	A device that runs a Web browser, acts as an interface to users for accessing Web information.
Web server	A device that responds to Web client requests.
Web-based simulation	A method of engaging a typical simulation application via the WWW technologies.
World Wide Web (WWW)	One of the Internet applications, which enables users to access documents and information from any locations.